D2:U.M65

```
                    1000      .TITLE "UPN_DOWN - for Atari systems"
                    1020 ;
                    1040 ; U.M65
                    1060 ;
                    1080 ;;   .OPT NO LIST
                    1100      .SET 3,0
     =4000          1120 ORG =   $4000
0000                1140      *=  ORG
4000                1160 TIMRON *= *+1    ; flag to enable timer
4001                1180 TICS *= *+1      ; 60ths sec counter
4002                1200 PSCOR0 *= *+1    ; player's score
4003                1220 PSCOR1 *= *+1
4004                1240 PSCOR2 *= *+1
4005                1260 ROUND *= *+1
4006                1280 SCRFLG2 *= *+1   ; flags VBI to scroll playfield
4007                1300 SNDCTL *= *+1    ; controls sound on/off
4008                1320 TRUCKX *= *+1
4009                1340 TRUCKY *= *+1
400A                1360 DSCROL *= *+1
400B                1380 SCRCTR *= *+1
400C                1400 SCRSPD *= *+1
400D                1420 SCROLF *= *+1
400E                1440 KX   *=  *+1
400F                1460 KY   *=  *+1
4010                1480 CX   *=  *+1
4011                1500 CY   *=  *+1
4012                1520 OLDVPOS *= *+1
4013                1540 OLDHPOS *= *+1
4014                1560 CHINDX *= *+1
4015                1580 CURCHAR *= *+1
4016                1600 SAVBUF *= *+40
403E                1620 DSPTBL *= *+223 ; table of address ptrs to screen (low)
411D                1640 DSPTBH *= *+223 ;                "              (high)
41FC                1660 M4TBL *= *+16
420C                1680 CHARH *= *+64   ; addr. high for char index
424C                1700 CHARL *= *+64   ; addr. low for char index
428C                1720 ROWBUF *= *+10
4296                1740 CARLFT *= *+1
4297                1760 PLYRDAT *= *+16
42A7                1780 LINCNT *= *+1
42A8                1800 BUGX *= *+1
42A9                1820 BUGY *= *+1
42AA                1840 DLIST *= *+248
43B6                1860 PFDATA *= *+1024
   =0370            1880 HMEMSIZ = 880
47B6                1900 HMEM *= *+HMEMSIZ ; header mem
   =004D            1920 ATRACT = $4D
   =02FC            1940 CH  =   $02FC
   =02F2            1960 CH1 =   $02F2
4B26 000A141E       1980 M10TBL .BYTE 0,10,20,30,40,50
4B2A 2832
4B2C 00050A0F       2000 M5TBL .BYTE 0,5,10,15,20,25,30,35
4B30 14191E23
4B34 282D3237       2020       .BYTE 40,45,50,55,60,65,70,75
4B38 3C41464B
4B3C                2040 M3TBL *= *+80
```

listing circa ~1983
Donated by
Charlie Kuelz
5-15-18

```
4B8C 0000       2060 CHRREF .WORD 0,0,0,0
```



```
4B8E 0000
4B90 0000
4B92 0000
      =D40B     2080 VCOUNT = $D40B
      =D20A     2100 RANDOM = $D20A
      =0224     2120 VVBLKD = $0224
      =0222     2140 VVBLKI = $0222
      =E462     2160 XITVBV = $E462
      =9000     2180 CHARS = $9000   ; start of char data (size= 4k)
      =7E70     2200 SMEM = $8000-400 ; screen mem. (size=4480 bytes), must
                                        align to 4k
      =026C     2220 PFLEN = 620     ; # of Playfield chars
      =D01F     2240 CONSOL = $D01F
      =D40E     2260 NMIEN = $D40E
      =D40A     2280 WSYNC = $D40A
      =0230     2300 SDLSTL = $0230
      =0231     2320 SDLSTH = $0231
      =00D4     2340 R    =   $D4
      =00CA     2360 SCROLC = $CA
      =7000     2380 PMEM =   $7000   ; player/missile mem. (first 768 not used)
      =D000     2400 HPOSP0 = $D000
      =D001     2420 HPOSP1 = $D001
      =D002     2440 HPOSP2 = $D002
      =D003     2460 HPOSP3 = $D003
      =D004     2480 HPOSM0 = $D004
      =D005     2500 HPOSM1 = $D005
      =D006     2520 HPOSM2 = $D006
      =D007     2540 HPOSM3 = $D007
      =D407     2560 PMBASE = $D407
      =02C0     2580 PCOLR0 = $02C0
      =02C1     2600 PCOLR1 = $02C1
      =02C2     2620 PCOLR2 = $02C2
      =02C3     2640 PCOLR3 = $02C3
      =D012     2660 COLPM0 = $D012
      =D013     2680 COLPM1 = $D013
      =D014     2700 COLPM2 = $D014
      =D015     2720 COLPM3 = $D015
      =026F     2740 GPRIOR = $026F
      =D01B     2760 PRIOR = $D01B
      =D008     2780 SIZEP0 = $D008
      =D009     2800 SIZEP1 = $D009
      =D00A     2820 SIZEP2 = $D00A
      =D00B     2840 SIZEP3 = $D00B
      =E000     2860 OSCHARS = $E000
      =02F4     2880 CHBAS = $02F4
      =D409     2900 CHBASE = $D409
      =7300     2920 MO   =   PMEM+768
      =7400     2940 P0   =   PMEM+1024
      =7500     2960 P1   =   PMEM+1024+256
      =7600     2980 P2   =   PMEM+1024+512
      =7700     3000 P3   =   PMEM+1024+768
      =022F     3020 SDMCTL = $022F
      =D400     3040 DMACTL = $D400
      =D01D     3060 GRACTL = $D01D
      =7800     3080 IPF =    $7800   ; start of playfield data
      =D405     3100 VSCROL = $D405
      =0012     3120 RTCLOK = $12
                3140 ;
```

Moore Business Forms, Inc. #1212R

```
=7800           3080 IPF =    $7800    ; start of playfield data
=D405           3100 VSCROL = $D405
=0012           3120 RTCLOK = $12
                3140 ;
```



```
                3160 ;
4B94            3180      *=  $4C00
4C00            3190 START
4C00 A200       3200      LDX #0      ; init stack
4C02 9A         3240      TXS
4C03 202C50     3260      JSR IVARS
4C06 20994F     3280      JSR ITABLES
4C09 20C64C     3300      JSR IPFDATA
4C0C 20034F     3320      JSR IDLIST
4C0F 206255     3340      JSR ITCHARS
4C12 207C54     3360      JSR IHMEM
4C15 20A44E     3380      JSR ISMEM
4C18 20FB4C     3400      JSR IVBI
4C1B 20884C     3420      JSR IPMEM
4C1E 204064     3440      JSR IOBJTB  ; init OBJTBL
4C21 200569     3460      JSR ISFX    ; init sound effects
4C24 20E06F     3480      JSR IFLAGS
4C27 A900       3500      LDA #0
4C29 8D586B     3520      STA TYPREF
4C2C A901       3540      LDA #1
4C2E 8D596B     3560      STA TYPREF+1
4C31 20A56A     3580      JSR DATEXP  ; expand data
4C34 A900       3600      LDA #0
4C36 8D0240     3620      STA PSCOR0  ; init player's score
4C39 8D0340     3640      STA PSCOR1
4C3C 8D0440     3660      STA PSCOR2
4C3F 8D0540     3680      STA ROUND
4C42 200966     3700      JSR NEWRND  ; setup next round
4C45 A901       3720      LDA #1      ; sound on
4C47 8D0740     3740      STA SNDCTL
4C4A A240       3760      LDX #64     ; set car's X
4C4C 8EA842     3780      STX BUGX
4C4F A946       3800      LDA #70
4C51 8DDD5A     3820      STA OLDBUGY
4C54 A030       3840      LDY #48     ; set car's Y
4C56 8CA942     3860      STY BUGY
4C59 A900       3880      LDA #0
4C5B 8DD95A     3900      STA BUGDIR
4C5E A907       3920      LDA #7
4C60 8DDE5A     3940      STA OLDBDIR
4C63 20BA5E     3960      JSR SHOCAR
4C66 A905       3980      LDA #5
4C68 8D9642     4000      STA CARLFT
4C6B 203054     4020      JSR ECAR
                4040 ;
4C6E A900       4060      LDA #0
4C70 8D0C40     4080      STA SCRSPD
4C73 8D1440     4100      STA CHINDX
                4120 ;
4C76 A93C       4140      LDA #$3C    ; enable joystick portA
4C78 8D02D3     4160      STA PACTL
4C7B A980       4180      LDA #128
4C7D 8DE05A     4200      STA JMPNDX
4C80 A900       4220      LDA #0
4C82 8DDA5A     4240      STA BPTHCT
4C85 4C204D     4260      JMP MAIN10
                4360 ;
```

Moore Business Forms, Inc. sv 14128

```
4380 ;
4C88                4400 IPMEM
4C88 A970           4420     LDA # >PMEM ; init PMBASE
4C8A 8D07D4         4440     STA PMBASE
4C8D A000           4460     LDY #0      ; clear PMEM
4C8E A205           4480     LDX #5      ; 5*256 bytes
4C91 A900           4500     LDA # <APMEM+768U ; set pointer to PMEM active area
4C93 85D4           4520     STA R+0
4C95 A973           4540     LDA # >APMEM+768U
4C97 85D5           4560     STA R+1
4C99 A900           4580     LDA #0
4C9B                4600 IPME10
4C9B 91D4           4620     STA (R+0),Y
4C9D C8             4640     INY
4C9E D0FB           4660     BNE IPME10
4CA0 CA             4680     DEX
4CA1 F005           4700     BEQ IPME20
4CA3 E6D5           4720     INC R+1
4CA5 4C9B4C         4740     JMP IPME10
4CA8                4760 IPME20
4CA8 A931           4780     LDA #$31    ; enable 3rd col. combo, missiles as 5th
                             plyr
4CAA 8D6F02         4800     STA GPRIOR
4CAD A900           4820     LDA #0      ; 1x size for all
4CAF 8D08D0         4840     STA SIZEP0
4CB2 8D09D0         4860     STA SIZEP1
4CB5 8D0AD0         4880     STA SIZEP2
4CB8 8D0BD0         4900     STA SIZEP3
4CBB A93E           4920     LDA #$3E
4CBD 8D2F02         4940     STA SDMCTL
4CC0 A903           4960     LDA #3
4CC2 8D1DD0         4980     STA GRACTL
4CC5 60             5000     RTS
                    5020 ;
                    5040 ;
4CC6                5060 IPFDATA
4CC6 A96C           5080     LDA # <PFLEN ; set # bytes to write
4CC8 85D4           5100     STA R+0
4CCA A902           5120     LDA # >PFLEN
4CCC 85D5           5140     STA R+1
4CCE A9B6           5160     LDA # <PFDATA ; set dest. ptr
4CD0 85D6           5180     STA R+2
4CD2 A943           5200     LDA # >PFDATA
4CD4 85D7           5220     STA R+3
4CD6 A900           5240     LDA # <IPF  ; set source ptr
4CD8 85D8           5260     STA R+4
4CDA A978           5280     LDA # >IPF
4CDC 85D9           5300     STA R+5
4CDE A000           5320     LDY #0
4CE0                5340 IPFD10
4CE0 B1D8           5360     LDA (R+4),Y
4CE2 91D6           5380     STA (R+2),Y
4CE4 C8             5400     INY
4CE5 C000           5420     CPY #0
4CE7 D0F7           5440     BNE IPFD10
4CE9 E6D9           5460     INC R+5
4CEB E6D7           5480     INC R+3
4CED C6D5           5500     DEC R+1
4CEF 10EF           5520     BPL IPFD10
```

Moore Business Forms, Inc. sv 14128

```
4CE9 E6D9       5460     INC R+5
4CEB E6D7       5480     INC R+3
4CED C6D5       5500     DEC R+1
4CEF 10EF       5520     BPL IPFD10
```



```
4CF1            5540 IPFD20
4CF1 B1D8       5560     LDA (R+4),Y
4CF3 91D6       5580     STA (R+2),Y
4CF5 C8         5600     INY
4CF6 C4D4       5620     CPY R+0
4CF8 D0F7       5640     BNE IPFD20
4CFA            5660 IPFD30
4CFA 60         5680     RTS
                5700 ;
                5720 ;
                5740 ;
                5760 ;
4CFB            5780 IVBI
4CFB AD0ED4     5800     LDA NMIEN
4CFE 293F       5820     AND #$3F    ; disable DLI, VBI
4D00 8D0ED4     5840     STA NMIEN
4D03 A9AA       5860     LDA # <DLI1
4D05 8D0002     5880     STA $0200
4D08 A953       5900     LDA # >DLI1
4D0A 8D0102     5920     STA $0201
4D0D A990       5940     LDA # <VBINT
4D0F 8D2202     5960     STA VVBLKI
4D12 A94D       5980     LDA # >VBINT
4D14 8D2302     6000     STA VVBLKI+1
4D17 AD0ED4     6020     LDA NMIEN
4D1A 0940       6040     ORA #$40
4D1C 8D0ED4     6060     STA NMIEN
4D1F 60         6080     RTS
                6100 ;
                6120 ;
                6140 ;
                6160 ;
                6180 ;
4D20            6200 MAIN10
                6220 ;;  LDA #0       ; defeat attract
                6240 ;;  STA ATRACT
4D20 202669     6260     JSR SCROLL
4D23 20975C     6280     JSR PMOVE
4D26 204170     6300     JSR FLGSCR
4D29 205863     6320     JSR MOVE
4D2C A900       6330     LDA #0       ; type to init
4D2E 8DD865     6332     STA TPNDX
4D31 20A564     6340     JSR TINIT
4D34 20076F     6360     JSR FLGHIT
4D37 205E50     6380     JSR ENDRND
                6400 ;
                6420 ; Check for trucks hitting player
                6440 ;
4D3A ADE05A     6460     LDA JMPNDX  ; only do this if not jumping
4D3D 1024       6480     BPL MAIN15
4D3F A201       6490     LDX #OBJMAX-1
4D41 20D965     6500     JSR COLISN
4D44 901D       6520     BCC MAIN15
4D46 A9FF       6540     LDA #-1     ; cancel truck
4D48 9DEC62     6560     STA TYPE,X
4D4B 204B64     6580     JSR ERASE   ; erase truck
4D4E 205C66     6600     JSR VWDIS
```

Moore Business Forms, Inc. sv 1412RH

```
4D51 CE9642     6620     DEC CARLFT
```



```
4D54 AD9642     6640     LDA CARLFT
4D57 C906       6660     CMP #6
4D59 9005       6680     BCC MAIN08
4D5B A905       6700     LDA #5
4D5D 8D9642     6720     STA CARLFT
4D60            6740 MAIN08
4D60 203054     6760     JSR ECAR    ; show extra cars
                6780 ;
4D63            6800 MAIN15
4D63 AD1FD0     6820     LDA CONSOL
4D66 2902       6840     AND #2
4D68 D008       6860     BNE MAIN20
4D6A A900       6880     LDA #0
4D6C 8D0C40     6900     STA SCRSPD
4D6F 4C8D4D     6920     JMP MAIN50
4D72            6940 MAIN20
4D72 AD1FD0     6960     LDA CONSOL
4D75 2901       6980     AND #1
4D77 D008       7000     BNE MAIN30
4D79 A9FF       7020     LDA #-1
4D7B 8D0C40     7040     STA SCRSPD
4D7E 4C8D4D     7060     JMP MAIN50
4D81            7080 MAIN30
4D81 AD1FD0     7100     LDA CONSOL
4D84 2904       7120     AND #4
4D86 D005       7140     BNE MAIN50
4D88 A901       7160     LDA #1
4D8A 8D0C40     7180     STA SCRSPD
                7200 ;
4D8D            7220 MAIN50
4D8D 4C204D     7240     JMP MAIN10
                7260 ;
                7280 ;
4D90            7300 VBINT
                7320 ;
4D90 A21F       7340     LDX #31
4D92            7360 VBIN05
4D92 B5D4       7380     LDA R+0,X   ; save scratchpad regs.
4D94 9D1640     7400     STA SAVBUF,X
4D97 CA         7420     DEX
4D98 10F8       7440     BPL VBIN05
                7460 ;
                7480 ; Set colors for header
                7500 ;
4D9A A90A       7520     LDA #$0A    ; lum
4D9C 8D17D0     7540     STA $D017
4D9F A910       7560     LDA #$10    ; hue
4DA1 8D18D0     7580     STA $D018
4DA4 A9AA       7600     LDA # <DLI1 ; set DLI vect. for bonus page
4DA6 8D0002     7620     STA $0200
4DA9 A953       7640     LDA # >DLI1
4DAB 8D0102     7660     STA $0201
                7680 ;
4DAE AD3002     7700     LDA SDLSTL  ; shadow registers
4DB1 8D02D4     7720     STA $D402
4DB4 AD3102     7740     LDA SDLSTH
4DB7 8D03D4     7760     STA $D403
```

Moore Business Forms, Inc. 3V 141ZKH

```
4DB1 8D02D4     7720     STA $D402
4DB4 AD3102     7740     LDA SDLSTH
4DB7 8D03D4     7760     STA $D403
4DBA AD2F02     7780     LDA SDMCTL
```



```
4DBD 8D00D4     7800     STA DMACTL
4DC0 ADF402     7820     LDA CHBAS
4DC3 8D09D4     7840     STA CHBASE
4DC6 A900       7860     LDA #0
4DC8 8D1AD0     7880     STA $D01A
4DCB ADC002     7900     LDA PCOLR0
4DCE ADC102     7920     LDA PCOLR1
4DD1 8D13D0     7940     STA COLPM1
4DD4 ADC202     7960     LDA PCOLR2
4DD7 8D14D0     7980     STA COLPM2
4DDA ADC302     8000     LDA PCOLR3
4DDD 8D15D0     8020     STA COLPM3
4DE0 AD6F02     8040     LDA GPRIOR
4DE3 8D1BD0     8060     STA PRIOR
4DE6 AD3202     8080     LDA SSKCTL
4DE9 8D0FD2     8100     STA SKCTL
                8120 ;
                8140 ; Setup header P/M flags
                8160 ;
4DEC AC3C70     8180     LDY HFPOS  ; HPOS
4DEF 8C07D0     8200     STY HPOSM3
4DF2 C8         8220     INY
4DF3 C8         8240     INY
4DF4 8C06D0     8260     STY HPOSM2
4DF7 C8         8280     INY
4DF8 C8         8300     INY
4DF9 8C05D0     8320     STY HPOSM1
4DFC C8         8340     INY
4DFD C8         8360     INY
4DFE 8C04D0     8380     STY HPOSM0
4E01 AD3D70     8400     LDA HFPOS+1
4E04 8D00D0     8420     STA HPOSP0
4E07 AD3E70     8440     LDA HFPOS+2
4E0A 8D01D0     8460     STA HPOSP1
4E0D AD3F70     8480     LDA HFPOS+3
4E10 8D02D0     8500     STA HPOSP2
4E13 AD4070     8520     LDA HFPOS+4
4E16 8D03D0     8540     STA HPOSP3
                8560 ;
4E19 ADAE6F     8580     LDA FCOLOR  ; colors
4E1C 8D19D0     8600     STA $D019
4E1F ADAF6F     8620     LDA FCOLOR+1
4E22 8D12D0     8640     STA COLPM0
4E25 ADB06F     8660     LDA FCOLOR+2
4E28 8D13D0     8680     STA COLPM1
4E2B ADB16F     8700     LDA FCOLOR+3
4E2E 8D14D0     8720     STA COLPM2
4E31 ADB26F     8740     LDA FCOLOR+4
4E34 8D15D0     8760     STA COLPM3
                8780 ;
4E37            8800 VBIN20
4E37 AD0040     8820     LDA TIMRON  ; timer on?
4E3A F027       8840     BEQ VBIN50  ; no - branch
4E3C CE0140     8860     DEC TICS
4E3F D021       8880     BNE VBIN55
4E41 A93C       8900     LDA #60
4E43 8D0140     8920     STA TICS
```

Moore Business Forms, Inc. sv 1412RH

```
                        8940 ;
4E46 F8                 8960    SED
4E47 AD5450             8980    LDA TIMER
4E4A 18                 9000    CLC
4E4B 6901               9010    ADC #1
4E4D 8D5450             9012    STA TIMER
4E50 C960               9020    CMP #$60
4E52 900E               9040    BCC VBIN55
4E54 A900               9060    LDA #0
4E56 8D5450             9080    STA TIMER
4E59 AD5550             9100    LDA TIMER+1
4E5C 18                 9102    CLC
4E5D 6901               9110    ADC #1
4E5F 8D5550             9120    STA TIMER+1
4E62                    9122 VBIN55
4E62 D8                 9140    CLD
4E63                    9150 VBIN50
                        9160 ;
                        9180 ;
                        9200 ; Sounds
                        9220 ;
4E63 AD0740             9240    LDA SNDCTL  ; sound enabled?
4E66 F02C               9260    BEQ VBIN32  ; no - branch
4E68 207068             9280    JSR SFX     ; else do sounds
                        9281 ;
                        9282 ; BONUS PAGE
                        9283 ;
4E6B AD5C50             9300    LDA BPAGEF  ; playfield or bonus page display?
4E6E F024               9320    BEQ VBIN32  ; playfield - branch
4E70 A9F2               9340    LDA # <BDLI1 ; set dli vector
4E72 8D0002             9360    STA $0200
4E75 A952               9380    LDA # >BDLI1
4E77 8D0102             9400    STA $0201
4E7A A900               9420    LDA #0
4E7C 8D00D0             9421    STA HPOSP0
4E7F 8D01D0             9422    STA HPOSP1
4E82 8D02D0             9423    STA HPOSP2
4E85 8D03D0             9424    STA HPOSP3
4E88 8D04D0             9425    STA HPOSM0
4E8B 8D05D0             9426    STA HPOSM1
4E8E 8D06D0             9427    STA HPOSM2
4E91 8D07D0             9428    STA HPOSM3
                        9440 ;
                        9460 ; Restore scratchpad regs
                        9480 ;
4E94                    9500 VBIN32
4E94 A21F               9520    LDX #31
4E96                    9540 VBIN40
4E96 BD1640             9560    LDA SAVBUF,X
4E99 95D4               9580    STA R+0,X
4E9B CA                 9600    DEX
4E9C 10F8               9620    BPL VBIN40
                        9640 ;
4E9E 68                 9660    PLA
4E9F A8                 9680    TAY
4EA0 68                 9700    PLA
4EA1 AA                 9720    TAX
4EA2 68                 9740    PLA
4EA3 40                 9760    RTI
```

```
4EA0 68          9700      PLA
4EA1 AA          9720      TAX
4EA2 68          9740      PLA
4EA3 40          9760      RTI
```



```
                 9780 ;
                 9800 ; Init. the screen memory with 120 lines of PF
                 9820 ; pointed to by SCROL ptrs
                 9840 ;
4EA4             9860 ISMEM
                 9880 ; LDA NMIEN
                 9900 ; AND #$BF    ; disable VBI
                 9920 ; STA NMIEN
4EA4 AD0D40      9940      LDA SCROLF
4EA7 48          9960      PHA
4EA8 A5CA        9980      LDA SCROLC
4EAA 48          010000    PHA
4EAB A5CB        010020    LDA SCROLC+1
4EAD 48          010040    PHA
4EAE A970        010060    LDA #112     ; do 112 PF lines
4EB0 85DD        010080    STA R+9
4EB2 AE0A40      010100    LDX DSCROL   ; get pointer to SMEM
4EB5             010120 ISME10
4EB5 BD3E40      010140    LDA DSPTBL,X ; set (0,1) to SMEM begin
4EB8 85D4        010160    STA R+0
4EBA BD1D41      010180    LDA DSPTBH,X
4EBD 85D5        010200    STA R+1
4EBF 205B6A      010220    JSR PFLINE   ; write a PF line
                 010240 ;
4EC2 C6DD        010260    DEC R+9
4EC4 F032        010280    BEQ ISME30
                 010300 ;
4EC6 E8          010320    INX          ; update screen pointer
4EC7 EE0D40      010340    INC SCROLF
4ECA AD0D40      010360    LDA SCROLF
4ECD C910        010380    CMP #16
4ECF D0E4        010400    BNE ISME10
4ED1 A900        010420    LDA #0
4ED3 8D0D40      010440    STA SCROLF
4ED6 A5CA        010460    LDA SCROLC
4ED8 18          010480    CLC
4ED9 690A        010500    ADC #10
4EDB 85CA        010520    STA SCROLC
4EDD A5CB        010540    LDA SCROLC+1
4EDF 6900        010560    ADC #0
4EE1 85CB        010580    STA SCROLC+1
4EE3 C946        010600    CMP # >PFDATA+PFLEN
4EE5 D0CE        010620    BNE ISME10
4EE7 A5CA        010640    LDA SCROLC
4EE9 C922        010660    CMP # <PFDATA+PFLEN
4EEB D0C8        010680    BNE ISME10
4EED A9B6        010700    LDA # <PFDATA ; reset
4EEF 85CA        010720    STA SCROLC
4EF1 A943        010740    LDA # >PFDATA
4EF3 85CB        010760    STA SCROLC+1
4EF5 4CB54E      010780    JMP ISME10
4EF8             010800 ISME30
4EF8 68          010820    PLA
4EF9 85CB        010840    STA SCROLC+1
4EFB 68          010860    PLA
4EFC 85CA        010880    STA SCROLC
4EFE 68          010900    PLA
```

Moore Business Forms, Inc. SV 14126H

```
4EFF 8D0D40     010920   STA SCROLF
```



```
                010940 ;
                010960 ; LDA NMIEN
                010980 ; ORA #$40
                011000 ; STA NMIEN
4F02 60         011020   RTS
                011040 ;
                011060 ; Init. the DLIST
                011080 ;
4F03            011100 IDLIST
4F03 A970       011120   LDA #$70
4F05 8DAA42     011140   STA DLIST
4F08 8DAB42     011160   STA DLIST+1
4F0B 8DAC42     011180   STA DLIST+2
4F0E A942       011200   LDA #$42     ; 1 line mode 2+load
4F10 8DAD42     011220   STA DLIST+3
4F13 A9B6       011240   LDA # <HMEM
4F15 8DAE42     011260   STA DLIST+4
4F18 A947       011280   LDA # >HMEM
4F1A 8DAF42     011300   STA DLIST+5
4F1D A982       011320   LDA #$82     ; 1 line mode 2 + DLI
4F1F 8DB042     011340   STA DLIST+6
4F22 A900       011360   LDA #$00     ; blank 1 scan line
4F24 8DB142     011380   STA DLIST+7
4F27 A90E       011400   LDA #$0E     ; 15 lines mode E
4F29 A200       011420   LDX #0
4F2B            011440 IDLI10
4F2B 9DB242     011460   STA DLIST+8,X
4F2E E8         011480   INX
4F2F E00F       011500   CPX #15
4F31 D0F8       011520   BNE IDLI10
4F33 A98E       011540   LDA #$8E     ; 1 line mode E + DLI (top flag DLI)
4F35 8DC142     011560   STA DLIST+23
                011580 ;
4F38 A98D       011600   LDA #$8D     ; 1 line mode D + DLI (blank line to
                                       separate header and playfield)
4F3A 8DC242     011620   STA DLIST+24
                011640 ;
4F3D A200       011660   LDX #0
4F3F A96D       011680   LDA #$6D
4F41            011700 IDLI30
4F41 9DC342     011720   STA DLIST+25,X ; 80 lines mode D w/load + vert. scroll
                                          enable
4F44 E8         011740   INX
4F45 E8         011760   INX
4F46 E8         011780   INX
4F47 E0F0       011800   CPX #240
4F49 D0F6       011820   BNE IDLI30
                011840 ;
4F4B 8DB043     011860   STA DLIST+262
4F4E A941       011880   LDA #$41     ; jump and wait instr.
4F50 8DB343     011900   STA DLIST+265
4F53 A9AA       011920   LDA # <DLIST
4F55 8DB443     011940   STA DLIST+266
4F58 A942       011960   LDA # >DLIST
4F5A 8DB543     011980   STA DLIST+267
4F5D A900       012000   LDA #0       ; reset pointer for DLIST scroll
4F5F 8D0A40     012020   STA DSCROL    ; to top of SMEM
4F62 8D0D40     012040   STA SCROLF
```

```
4F5A 8DB543      011980   STA DLIST+267
4F5D A900        012000   LDA #0       ; reset pointer for DLIST scroll
4F5F 8D0A40      012020   STA DSCROL   ; to top of SMEM
4F62 8D0D40      012040   STA SCROLF
```



```
4F65 A9B6        012060   LDA # <PFDATA
4F67 85CA        012080   STA SCROLC
4F69 A943        012100   LDA # >PFDATA
4F6B 85CB        012120   STA SCROLC+1
4F6D 207B4F      012140   JSR CHGDSP   ; set DLIST pointers
4F70 A9AA        012160   LDA # <DLIST ; select DLIST
4F72 8D3002      012180   STA SDLSTL
4F75 A942        012200   LDA # >DLIST
4F77 8D3102      012220   STA SDLSTH
4F7A 60          012240   RTS
                 012260 ;
                 012280 ; Write the display list pointers
                 012300 ; using the value in DSCROL to
                 012320 ; determine tne first line.
4F7B             012340 CHGDSP
4F7B AD0A40      012360   LDA DSCROL   ; use to index first line
4F7E 18          012380   CLC          ; first line is +16 to leave room for
                                       "before" buffer
4F7F 6910        012400   ADC #16
4F81 AA          012420   TAX
4F82 A000        012440   LDY #0
4F84             012460 MOVD10
4F84 BD3E40      012480   LDA DSPTBL,X ; get low byte of pointer
4F87 99C442      012500   STA DLIST+26,Y ; write to DLIST
4F8A BD1D41      012520   LDA DSPTBH,X ;    high byte
4F8D 99C542      012540   STA DLIST+27,Y
4F90 E8          012560   INX          ; next DSPTBL value
4F91 C8          012580   INY          ; next DLIST pointer
4F92 C8          012600   INY
4F93 C8          012620   INY
4F94 C0F0        012640   CPY #240     ; 80 lines
4F96 D0EC        012660   BNE MOVD10
4F98 60          012680   RTS
                 012700 ; INIT THE TABLES
                 012720 ;
                 012740 ; Setup DSPTBL,DSPTBH with pointer
                 012760 ; values for the display list
                 012780 ; scroll.  Though there are 80
                 012800 ; There are 223 entries in DSPTB,  112 to
                 012820 ; handle the addresses of the 80 displayed mode
                 012840 ; lines, 16 for "before" and 16 for "after" buffers,
                 012860 ; 111 to extend the table so that even
                 012880 ; when indexing starting at line #111
                 012900 ; there will be no need to wraparound the index
                 012920 ;
                 012940 ;
4F99             012960 ITABLES
4F99 A270        012980   LDX #112   ; do first 80 pointers
4F9B A93E        013000   LDA # <DSPTBL ; set pointer to DSPTBL
4F9D 85D6        013020   STA R+2
4F9F A940        013040   LDA # >DSPTBL
4FA1 85D7        013060   STA R+3
4FA3 A000        013080   LDY #0       ; index to DSPTBL
4FA5             013100 ITAB05
4FA5 A970        013120   LDA # <SMEM ; set initial pointer value
4FA7 85D4        013140   STA R+0
4FA9 A97E        013160   LDA # >SMEM
```

Moore Business Forms, Inc. s/ 1412RH

```
4FAB 85D5       013180   STA R+1
```



```
                013200 ITAB10
4FAD            013220   LDA R+1      ; store a pointer
4FAD A5D5       013240   STA DSPTBH,Y ; high byte
4FAF 991D41     013260   LDA R+0
4FB2 A5D4       013280   STA DSPTBL,Y ; low byte
4FB4 993E40     013300   INY
4FB7 C8         013320   DEX
4FB8 CA         013340   BEQ ITAB20
4FB9 F01D       013360 ;
                013380 ; LDA R+1      ; save high nybble for 4k boundary check
                013400 ; AND #$F0
                013420 ; STA R+4
                013440 ; LDA R+0
                013460   CLC          ; update pointer value
4FBB 18         013480   ADC #40      ; 40 bytes/mode line
4FBC 6928       013500   STA R+0
4FBE 85D4       013520   BCC ITAB10
4FC0 90EB       013540   INC R+1
4FC2 E6D5       013560   JMP ITAB10 ; next pointer
4FC4 4CAD4F     013580 ;
                013600 ; Check for 4k boundary crossing
                013620 ;
4FC7            013640 ITAB15
4FC7 A5D5       013660   LDA R+1
4FC9 29F0       013680   AND #$F0
4FCB C5D8       013700   CMP R+4
4FCD F0DE       013720   BEQ ITAB10  ; ok - branch
4FCF 85D5       013740   STA R+1     ; else force to 4k
4FD1 A900       013760   LDA #$00
4FD3 85D4       013780   STA R+0
4FD5 4CAD4F     013800   JMP ITAB10
                013820 ;
                013840 ; do last 111 pointer values
                013860 ;
4FD8            013880 ITAB20
4FD8 A5D6       013900   LDA R+2      ; is this the first time here?
4FDA C93E       013920   CMP # <DSPTBL
4FDC D00E       013940   BNE ITAB30  ; no - already did this, branch
4FDE 18         013960   CLC         ; move pointer to DSPTBL
4FDF 6970       013980   ADC #112
4FE1 85D6       014000   STA R+2
4FE3 9002       014020   BCC ITAB28
4FE5 E6D7       014040   INC R+3
4FE7            014060 ITAB28
4FE7 A26F       014080   LDX #111    ; 111 more values
4FE9 4CA54F     014100   JMP ITAB05
                014120   ;
                014140 ; init CHARH,CHARL index to char
                014160 ; addresses.
                014180 ;
4FEC            014200 ITAB30
4FEC A900       014220   LDA # <CHARS ; init first addr to CHARS start
4FEE 85D4       014240   STA R+0
4FF0 A990       014260   LDA # >CHARS
4FF2 85D5       014280   STA R+1
4FF4 A000       014300   LDY #0
4FF6            014320 ITAB40
4FF6 A5D5       014340   LDA R+1      ; write a pointer
```

Moore Business Forms, Inc sv 1412RH

```
4FF0 A990       014260   LDA # >CHARS
4FF2 85D5       014280   STA R+1
4FF4 A000       014300   LDY #0
4FF6            014320 ITAB40
4FF6 A5D5       014340   LDA R+1     ; write a pointer
```



```
4FF8 990C42     014360   STA CHARH,Y
4FFB A5D4       014380   LDA R+0
4FFD 994C42     014400   STA CHARL,Y
                014420 ;
5000 C8         014440   INY
5001 C040       014460   CPY #64     ; do for # chars in CHARS
5003 F00C       014480   BEQ ITAB50
                014500 ;
5005 18         014520   CLC
5006 6940       014540   ADC #64
5008 85D4       014560   STA R+0
500A 90EA       014580   BCC ITAB40
500C E6D5       014600   INC R+1
500E 4CF64F     014620   JMP ITAB40
5011            014640 ITAB50
                014660 ;
                014680 ; init mult. by 4 table for fine
                014700 ; scroll add in calc.
                014720 ;
5011 A93C       014740   LDA #60
5013 A00F       014760   LDY #15
5015 38         014780   SEC         ; set for subt.
5016            014800 ITAB60
5016 99FC41     014820   STA M4TBL,Y
5019 E904       014840   SBC #4
501B 88         014860   DEY
501C 10F8       014880   BPL ITAB60
                014900 ;
                014920 ; init *3 table
                014940 ;
501E A9F0       014960   LDA #80*3   ; 80 entries
5020 A24F       014980   LDX #79
5022 38         015000   SEC
5023            015020 ITAB70
5023 E903       015040   SBC #3
5025 9D3C4B     015060   STA M3TBL,X
5028 CA         015080   DEX
5029 10F8       015100   BPL ITAB70
                015120 ;
502B 60         015140   RTS
                015160 ;
                015180 ; INIT THE VARIABLES
                015200 ;
502C            015220 IVARS
502C A900       015240   LDA #0
502E 8DE471     015260   STA FTOPDLI
5031 8D5C50     015270   STA BPAGEF
                015280 ;
5034 A950       015300   LDA #FDLY   ; init flag delay counter
5036 8D066F     015320   STA FDLYCT
5039 A900       015340   LDA #0       ; no flag toggle
503B 8D046F     015360   STA FTOGL
                015380 ;
503E A201       015400   LDX #OBJMAX-1
5040            015420 IVAR10
5040 A9FF       015440   LDA #$FF
5042 9D4663     015460   STA OVRLAY,X
```

Moore Business Forms, Inc. sv 1412RH

```
5045 9DEC62   .  015480   STA TYPE,X
```



```
5048 A900      015500   LDA #0
504A 9D4F63    015520   STA DRAWN,X
504D 9D2B63    015540   STA PATH,X
5050 CA        015560   DEX
5051 10ED      015580   BPL IVAR10
               015600 ;
5053 60        015620   RTS
               015640 ;
               015660 ;
               015680   .INCLUDE #D2:BPAGE.M65
               1000     .PAGE "BPAGE.M65"
```

Moore Business Forms, Inc. sv 1412RH

```
                        1020 ;
                        1040 ; BONUS PAGE ROUTINES
                        1060 ;
5054                    1080 TIMER *= *+1
5055                    1100 INCTR *= *+1
5056                    1120 FLASH *= *+1
5057                    1140 FLSHCT *= *+1
5058                    1160 CMSGCT *= *+1
5059                    1180 CFADE *= *+1
505A                    1200 FADECT *= *+1
505B                    1220 CMSGDN *= *+1
505C                    1240 BPAGEF *= *+1
     =0064              1260 CMSGDY = 100
     =00C8              1280 FLSDLY = 200
     =0014              1300 INCDLY = 20
     =0001              1320 CINC =  1       ; inc on CONGRAT's rainbow colors
505D                    1340 CON *=  *+1
                        1360 ;
                        1380 ; CHECK FOR END OF ROUND
                        1400 ;
505E                    1420 ENDRND
505E AD036F             1440     LDA FLGCNT  ; any flags left?
5061 F001               1460     BEQ ENDR10
5063 60                 1480     RTS
                        1500 ;
5064                    1520 ENDR10
5064 A900               1540     LDA #0      ; stop timer
5066 8D0040             1560     STA TIMRON
                        1580 ;
5069 20BC66             1600     JSR ALLFLGS ; play tune
506C                    1620 ENDR20
506C A200               1640     LDX #0
506E A005               1660     LDY #5
5070                    1680 ENDR22
5070 CA                 1700     DEX         ; delay
5071 D0FD               1720     BNE ENDR22
5073 88                 1740     DEY
5074 D0FA               1760     BNE ENDR22
                        1770 ;
5076 AD0AD2             1780     LDA RANDOM  ; get flag #
5079 290F               1800     AND #$0F
507B C90A               1820     CMP #10
507D B0ED               1840     BCS ENDR20
507F AA                 1860     TAX
5080 BDAE6F             1880     LDA FCOLOR,X ; flip header flag
5083 4901               1900     EOR #1
5085 9DAE6F             1920     STA FCOLOR,X
5088 20E571             1940     JSR CHGFLG
                        1960 ;
508B ADEE68             1980     LDA VPTR1+1 ; tune finished?
508E 0DE668             2000     ORA VOICE1+1
5091 D0D9               2020     BNE ENDR20  ; no - branch
                        2040 ;
5093 209950             2060     JSR BPAGE   ; do bonus page
5096 4C004C             2080     JMP START
                        2100 ;
                        2120 ; DO THE BONUS PAGE
```

Moore Business Forms, Inc. Form 1174

```
                        2140 ;
                        2160 BPAGE
5099 208B66             2180     JSR BSOUND  ; do tune
5099 A928               2200     LDA #40
509E                    2220 BPAG10
509E CD0BD4             2240     CMP VCOUNT  ; sync to screen
50A1 90FB               2260     BCC BPAG10
                        2280 ;
50A3 A901               2300     LDA #1
50A5 8D5C50             2320     STA BPAGEF  ; flag bonus page for VBI
                        2340 ;
50A8 A20D               2360     LDX #13     ; overwrite UP'N DOWN title
50AA                    2380 BPAG22
50AA BDE251             2400     LDA TCHERA,X
50AD 9DC347             2420     STA HMEM+13,X
50B0 BDF051             2440     LDA TCHERA+14,X
50B3 9DEB47             2460     STA HMEM+53,X
50B6 CA                 2480     DEX
50B7 10F1               2500     BPL BPAG22
                        2520 ;
50B9 A200               2540     LDX #0
50BB                    2560 BPAG20
50BB BDFE51             2580     LDA BPDAT,X ; init bonus page header
50BE 9D0648             2600     STA HMEM+80,X
50C1 E8                 2620     INX
50C2 E08C               2640     CPX #140
50C4 D0F5               2660     BNE BPAG20
50C6 A900               2680     LDA #0
50C8 A24F               2700     LDX #79
50CA                    2720 BPAG25
50CA 9D9248             2740     STA HMEM+220,X ; clear last 80 bytes
50CD CA                 2760     DEX
50CE 10FA               2780     BPL BPAG25
                        2800 ;
50D0 AD5550             2820     LDA TIMER+1 ; get timer high byte
50D3 4A                 2840     LSR A
50D4 4A                 2860     LSR A
50D5 4A                 2880     LSR A
50D6 4A                 2900     LSR A
50D7 F003               2920     BEQ BPAG30
50D9 8DEF47             2940     STA HMEM+57
50DC                    2960 BPAG30
50DC AD5550             2980     LDA TIMER+1
50DF 290F               3000     AND #$0F
50E1 8DF047             3020     STA HMEM+58
50E4 AD5450             3040     LDA TIMER
50E7 4A                 3060     LSR A
50E8 4A                 3080     LSR A
50E9 4A                 3100     LSR A
50EA 4A                 3120     LSR A
50EB 8DF247             3140     STA HMEM+60
50EE AD5450             3160     LDA TIMER
50F1 290F               3180     AND #$0F
50F3 8DF347             3200     STA HMEM+61
                        3220 ;
50F6 20A452             3240     JSR IBDLST
                        3260 ;
                        3280 ; Set index according to timer
                        3300 ;
```

Moore Business Forms, Inc. sv 1412RH

```
50F6 20A452     3240      JSR IBDLST
                3260 ;
                3280 ; Set index according to timer
                3300 ;
```



```
50F9 A205       3320      LDX #5
50FB AD5550     3340      LDA TIMER+1
50FE D00F       3360      BNE BPAG50
5100 A949       3380      LDA #$49
5102 A204       3400      LDX #4
5104            3420 BPAG40
5104 CD5450     3440      CMP TIMER
5107 9006       3460      BCC BPAG50
5109 38         3480      SEC
510A E910       3500      SBC #$10
510C CA         3520      DEX
510D D0F5       3540      BNE BPAG40
                3560 ;
                3580 ; Set parameters for incrementing score
                3600 ;
510F            3620 BPAG50
510F A901       3640      LDA #$01    ; add increments of 100
5111 85DA       3660      STA R+6
5113 A900       3680      LDA #$00
5115 85D9       3700      STA R+5
5117 BD9E52     3720      LDA BINC,X  ; get # of score incs in R+7
511A 85DB       3740      STA R+7
                3760 ;
511C BDCA51     3780      LDA BMSGL,X
511F 85DC       3800      STA R+8
5121 BDD051     3820      LDA BMSGH,X
5124 85DD       3840      STA R+9
5126 BDD651     3860      LDA BDATL,X
5129 85DE       3880      STA R+10
512B BDDC51     3900      LDA BDATH,X
512E 85DF       3920      STA R+11
                3940 ;
5130 A914       3960      LDA #INCDLY
5132 8D5550     3980      STA INCTR
5135 A9C8       4000      LDA #FLSDLY
5137 8D5750     4020      STA FLSHCT
513A A901       4040      LDA #1
513C 8D5650     4060      STA FLASH
                4080 ;
513F A964       4140      LDA #CMSGDY
5141 8D5850     4160      STA CMSGCT
5144 A900       4180      LDA #0
5146 8D5D50     4200      STA CON
5149 8D5B50     4220      STA CMSGON
514C E005       4240      CPX #5
514E F010       4260      BEQ BPAG60
5150 A901       4280      LDA #1
5152 8D5B50     4300      STA CMSGON
                4320 ;
5155 A213       4340      LDX #19     ; write CONGRATULATIONS
5157            4360 BPAG54
5157 BD8A52     4380      LDA CONGRA,X
515A 9DBA48     4400      STA HMEM+260,X
515D CA         4420      DEX
515E 10F7       4440      BPL BPAG54
                4460 ;
5160            4480 BPAG60
```

Moore Business Forms, Inc. av 1412H

```
5160 A200       4500    LDX #0
```



```
5162            4520 BPAG61
5162 CA         4540    DEX
5163 D0FD       4560    BNE BPAG61
5165 A5DB       4580    LDA R+7
5167 F00F       4600    BEQ BPAG70  ; all done - branch
5169 CE5550     4620    DEC INCTR
516C D00A       4640    BNE BPAG70
516E A914       4660    LDA #INCDLY
5170 8D5550     4680    STA INCTR
5173 20BB5F     4700    JSR ADDP1
5176 C6DB       4720    DEC R+7
5178            4740 BPAG70
5178 CE5750     4760    DEC FLSHCT  ; flash point bonus message?
517B D027       4780    BNE BPAG92  ; not time yet - branch
517D A9C8       4800    LDA #FLSDLY
517F 8D5750     4820    STA FLSHCT
5182 A013       4840    LDY #19
5184 AD5650     4860    LDA FLASH
5187 F00F       4880    BEQ BPAG90
5189 A900       4900    LDA #0      ; erase message
518B            4920 BPAG80
518B 91DC       4940    STA (R+8),Y
518D 88         4960    DEY
518E 10FB       4980    BPL BPAG80
5190 A900       5000    LDA #0
5192 8D5650     5020    STA FLASH
5195 4CA451     5040    JMP BPAG92
5198            5060 BPAG90
5198 B1DE       5080    LDA (R+10),Y ; write msg
519A 91DC       5100    STA (R+8),Y
519C 88         5120    DEY
519D 10F9       5140    BPL BPAG90
519F A901       5160    LDA #1
51A1 8D5650     5180    STA FLASH
                5200 ;
51A4            5220 BPAG92
51A4 CE5850     5240    DEC CMSGCT  ; flash CONGRATS
51A7 D00A       5260    BNE BPA100
51A9 A964       5280    LDA #CMSGDY
51AB 8D5850     5300    STA CMSGCT
                5320 ;; LDA CON
                5340 ;; EOR #1
51AE A901       5350    LDA #1      ; always on
51B0 8D5D50     5360    STA CON
                5380 ;
51B3            5400 BPA100
51B3 ADEE68     5420    LDA VPTR1+1 ; end of tune?
51B6 0DE668     5440    ORA VOICE1+1
51B9 0DF268     5460    ORA VPTR3+1
51BC 0DF468     5480    ORA VPTR4+1
51BF D09F       5500    BNE BPAG60  ; no - branch
51C1 A900       5520    LDA #0      ; reinit sound
51C3 8D0FD2     5540    STA SKCTL
51C6 8D3202     5560    STA SSKCTL
51C9 60         5580    RTS
                5600 ;
                5620 ;
                5640 ;
```

Moore Business Forms, Inc. sv 1412RH

```
51C9 60          5580      RTS
                 5600 ;
                 5620 ;
                 5640 ;
```



```
51CA             5660 BMSGL
51CA 1A2E4256    5680      .BYTE  <HMEM+100, <HMEM+120, <HMEM+140, <HMEM+160,
                                  <HMEM+180, <HMEM+200
51CE 6A7E
51D0             5700 BMSGH
51D0 48484848    5720      .BYTE  >HMEM+100, >HMEM+120, >HMEM+140, >HMEM+160,
                                  >HMEM+180, >HMEM+200
51D4 4848
51D6             5740 BDATL
51D6 12263A4E    5760      .BYTE  <BPDAT+20, <BPDAT+40, <BPDAT+60, <BPDAT+80,
                                  <BPDAT+100, <BPDAT+120
51DA 6276
51DC             5780 BDATH
51DC 52525252    5800      .BYTE  >BPDAT+20, >BPDAT+40, >BPDAT+60, >BPDAT+80,
                                  >BPDAT+100, >BPDAT+120
51E0 5252
                 5820 ;
51E2             5840 TCHERA
51E2 18181818    5860      .BYTE 24,24,24,24,24,25,14,26,11,24,24,24,24,24
51E6 18190E1A
51EA 0B181818
51EE 1818
51F0 18181818    5880      .BYTE 24,24,24,24,24,24,27,24,24,24,24,24,24,24
51F4 18181B18
51F8 18181818
51FC 1818
51FE             5900 BPDAT
51FE 00000000    5920      .SBYTE "                    "
5202 00000000
5206 00000000
520A 00000000
520E 00000000
5212 00000010    5940      .SBYTE "   0-19 SEC 20,000  "
5216 0D111900
521A 33252300
521E 12100C10
5222 10100000
5226 00001210    5960      .SBYTE "  20-29 SEC 10,000  "
522A 0D121900
522E 33252300
5232 11100C10
5236 10100000
523A 00001310    5980      .SBYTE "  30-39 SEC  5,000  "
523E 0D131900
5242 33252300
5246 00150C10
524A 10100000
524E 00001410    6000      .SBYTE "  40-49 SEC  3,000  "
5252 0D141900
5256 33252300
525A 00130C10
525E 10100000
5262 00001510    6020      .SBYTE "  50-59 SEC  1,000  "
5266 0D151900
526A 33252300
526E 00110C10
5272 10100000
```

Moore Business Forms, Inc. sv 1412RH

```
5276 00001610    6040      .SBYTE "   60 AND OVER ZERO
```



```
527A 00212E24
527E 002F3625
5282 32003A25
5286 322F0000
528A              6060 CONGRA
528A 001D232F     6080      .SBYTE " =CONGRATULATIONS=   "
528E 2E273221
5292 34352C21
5296 34292F2E
529A 331D0000
                  6100 ;
                  6120 ;
529E              6140 BINC
529E C864321E     6160      .BYTE 200,100,50,30,10,0
52A2 0A00
                  6180 ;
                  6200 ;
                  6220 ; INIT THE BONUS PAGE DISPLAY LIST
                  6240 ;
52A4              6260 IBDLST
52A4 A970         6280      LDA #$70
52A6 8DAA42       6300      STA DLIST
52A9 8DAB42       6320      STA DLIST+1
52AC 8DAC42       6340      STA DLIST+2
52AF A942         6360      LDA #$42    ; 1 line mode 2 + load
52B1 8DAD42       6380      STA DLIST+3
52B4 A9B6         6400      LDA # <HMEM
52B6 8DAE42       6420      STA DLIST+4
52B9 A947         6440      LDA # >HMEM
52BB 8DAF42       6460      STA DLIST+5
52BE A982         6480      LDA #$82    ; 1 line mode 2 + DLI
52C0 8DB042       6500      STA DLIST+6
52C3 A907         6520      LDA #$07    ; 8 lines mode 7
52C5 A207         6540      LDX #7
52C7              6560 IBDL10
52C7 9DB142       6580      STA DLIST+7,X
52CA CA           6600      DEX
52CB 10FA         6620      BPL IBDL10
52CD A287         6640      LDX #$87    ; 1 line mode 7 + DLI
52CF 8EB942       6660      STX DLIST+15
52D2 8DBA42       6680      STA DLIST+16
52D5 8DBB42       6700      STA DLIST+17
                  6720 ;
52D8 A941         6740      LDA #$41    ; jump and wait
52DA 8DBC42       6760      STA DLIST+18
52DD A9AA         6780      LDA # <DLIST
52DF 8DBD42       6800      STA DLIST+19
52E2 A942         6820      LDA # >DLIST
52E4 8DBE42       6840      STA DLIST+20
                  6860 ;
52E7 A9AA         6880      LDA # <DLIST
52E9 8D3002       6900      STA SDLSTL
52EC A942         6920      LDA # >DLIST
52EE 8D3102       6940      STA SDLSTH
                  6960 ;
52F1 60           6980      RTS
                  7000 ;
                  7020 ;
```

Moore Business Forms, Inc. sv 1412RH

```
52F1 60          6960 ;
                 6980      RTS
                 7000 ;
                 7020 ;
```



```
52F2             7040 BDLI1
52F2 48          7060      PHA
52F3 A90E        7080      LDA #$0E    ; white letters
52F5 8D0AD4      7100      STA WSYNC
52F8 8D16D0      7120      STA $D016
52FB A90C        7140      LDA # <BDLI2
52FD 8D0002      7160      STA $0200
5300 A953        7180      LDA # >BDLI2
5302 8D0102      7200      STA $0201
5305 A9E0        7220      LDA # >OSCHARS
5307 8D09D4      7240      STA CHBASE
530A 68          7260      PLA
530B 40          7280      RTI
                 7300 ;
530C             7320 BDLI2
530C 48          7340      PHA
530D 8A          7360      TXA
530E 48          7380      PHA
530F AD5D50      7400      LDA CON
5312 D006        7420      BNE B2DL05
5314 8D16D0      7440      STA $D016
5317 4C3853      7460      JMP B2DL20
                 7480 ;
531A             7500 B2DL05
531A AD5950      7520      LDA CFADE
531D 8D0AD4      7540      STA WSYNC
5320 8D16D0      7560      STA $D016
5323 18          7580      CLC
5324 A20F        7600      LDX #15
5326             7620 B2DL10
5326 6901        7640      ADC #CINC
5328 48          7660      PHA
5329 29FB        7680      AND #$FB
532B 8D0AD4      7700      STA WSYNC
532E 8D16D0      7720      STA $D016
5331 68          7740      PLA
5332 CA          7760      DEX
5333 D0F1        7780      BNE B2DL10
                 7800 ;
5335 8D5950      7820      STA CFADE
                 7840 ;
5338             7860 B2DL20
5338 68          7880      PLA
5339 AA          7900      TAX
533A 68          7920      PLA
533B 40          7940      RTI
                 015700    .INCLUDE #D2:HDRSUB.M65
                 1000      .PAGE "header subroutines"
```

Moore Business Forms, Inc. sv 1412RH

```
                        1020 ; HDRSUB.M65
                        1040 ; SUBROUTINES TO HANDLE THE
                        1060 ; HEADER DISPLAY FOR UPN DOWN
                        1080 ;
                        1100 ;
                        1120 ;
                        1140 ;
                        1160 ; *ADDSCOR*
                        1180 ; ADD THE 2 BYTE BCD VALUE IN R+5,R+6
                        1200 ; TO THE 3 BYTE BCD VALUE IN R+2,R+3,R+4
                        1220 ; AND DISPLAY AS A 6-DIGIT DECIMAL
                        1240 ; NUMBER AT SCREEN LOCATION POINTED
                        1260 ; TO BY R+0,R+1
                        1280 ;
533C                    1300 ADDSCOR
533C 8A                 1302     TXA
533D 48                 1304     PHA
533E F8                 1320     SED             ; decimal mode
533F A5D6               1340     LDA R+2         ; get LSB
5341 18                 1360     CLC
5342 65D9               1380     ADC R+5
5344 85D6               1400     STA R+2
5346 A5D7               1420     LDA R+3         ; middle byte
5348 65DA               1440     ADC R+6
534A 85D7               1460     STA R+3
534C A5D8               1480     LDA R+4
534E 6900               1500     ADC #0
5350 85D8               1520     STA R+4
5352 D8                 1530     CLD
5353                    1540 ADDS10
5353 A000               1560     LDY #0          ; ptr to dest.
5355 A202               1580     LDX #2          ; index to byte being written
                        1600 ;
                        1620 ; Find first non-zero
                        1640 ;
5357                    1650 ADDS18
5357 B5D6               1660     LDA R+2,X
5359 29F0               1680     AND #$F0
535B D018               1700     BNE ADDS20
535D A918               1720     LDA #24         ; space char
535F 91D4               1740     STA (R+0),Y
5361 C8                 1760     INY
5362 B5D6               1780     LDA R+2,X
5364 290F               1800     AND #$0F
5366 D016               1820     BNE ADDS24
5368 A918               1840     LDA #24
536A 91D4               1860     STA (R+0),Y
536C C8                 1880     INY
536D CA                 1900     DEX
536E 10E7               1920     BPL ADDS18
5370 E8                 1930     INX
5371 88                 1932     DEY
5372 4C7E53             1940     JMP ADDS24
                        1960 ;
5375                    1980 ADDS20
5375 B5D6               2000     LDA R+2,X       ; write high nybble
5377 4A                 2020     LSR A
5378 4A                 2040     LSR A
```

Moore Business Forms, Inc. sv 1412RH

```
                1960 ;
5375            1980 ADDS20
5375 B5D6       2000     LDA R+2,X    ; write high nybble
5377 4A         2020     LSR A
5378 4A         2040     LSR A
```



```
5379 4A         2060     LSR A
537A 4A         2080     LSR A
537B 91D4       2090     STA (R+0),Y
537D C8         2100     INY
537E            2110 ADDS24
537E B5D6       2260     LDA R+2,X    ; get byte
5380 290F       2280     AND #$0F     ; write low nybble
5382 91D4       2500     STA (R+0),Y
5384 C8         2520     INY
5385 CA         2540     DEX
5386 10ED       2560     BPL ADDS20
5388 68         2562     PLA
5389 AA         2564     TAX
538A 60         2600     RTS
                2620 ;
                2640 ;
                2660 ;
                2680 ; INIT THE PLAYER DATA TABLE
                2700 ;
538B            2720 IPLYRDT
538B A900       2740     LDA #0
538D A20F       2760     LDX #15
538F            2780 IPLY10
538F 9D9742     2800     STA PLYRDAT,X ; clear PLYRDAT data area
5392 CA         2820     DEX
5393 10FA       2840     BPL IPLY10
                2860 ;
5395 A9BC       2880     LDA # <HMEM+6 ; set pointer to PLYR1 store disp
5397 8D9742     2900     STA PLYRDAT
539A A947       2920     LDA # >HMEM+6
539C 8D9842     2940     STA PLYRDAT+1
539F A9E4       2960     LDA # <HMEM+46 ;      "        PLYR2      "
53A1 8D9F42     2980     STA PLYRDAT+8
53A4 A947       3000     LDA # >HMEM+46
53A6 8D9F42     3020     STA PLYRDAT+8
53A9 60         3040     RTS
                3060 ;
                3080 ; DLI routine to change colors
                3100 ; between header text (top) and
                3120 ; header graphics (bottom) areas
                3140 ;
53AA            3160 DLI1
53AA 48         3180     PHA
53AB A900       3200     LDA #$00     ; black
53AD 8D0AD4     3220     STA WSYNC
53B0 8D1AD0     3240     STA $D01A
53B3 A90E       3260     LDA #$0E     ; white
53B5 8D16D0     3280     STA $D016
53B8 A986       3300     LDA #$86     ; blue
53BA 8D17D0     3320     STA $D017
53BD A91E       3340     LDA #$1E     ; yellow
53BF 8D18D0     3360     STA $D018
53C2 A915       3380     LDA # <TFDLI ; set vector to next DLI
53C4 8D0002     3400     STA $0200
53C7 A954       3420     LDA # >TFDLI
53C9 8D0102     3440     STA $0201
53CC 68         3460     PLA
```

Moore Business Forms, Inc. sv 1412RH

```
53CD 40          3480      RTI
```



```
                 3500 ;
                 3520 ;
53CE             3540 DLI2
53CE 48          3560      PHA
53CF A9C6        3580      LDA #$C6     ; green
53D1 8D0AD4      3600      STA WSYNC
53D4 8D1AD0      3620      STA $D01A
53D7 A900        3640      LDA #0       ;black
53D9 8D18D0      3660      STA $D018
53DC A9F6        3680      LDA #$F6     ;brown
53DE 8D16D0      3700      STA $D016
53E1 A9A2        3720      LDA # <FDLI  ; set flag DLI vect.
53E3 8D0002      3740      STA $0200
53E6 A971        3760      LDA # >FDLI
53E8 8D0102      3780      STA $0201
                 3800 ;
                 3820 ; Restore VW colors
                 3840 ;
53EB A99A        3860      LDA #$9A     ; blue
53ED 8D12D0      3880      STA COLPM0
53F0 8D14D0      3900      STA COLPM2
53F3 A906        3920      LDA #$06     ; grey
53F5 8D13D0      3940      STA COLPM1
53F8 8D15D0      3960      STA COLPM3
                 3980 ;
                 4000 ; Restore VW position
                 4020 ;
53FB ADE55F      4040      LDA BUGH0
53FE 8D00D0      4060      STA HPOSP0
5401 ADE65F      4080      LDA BUGH1
5404 8D01D0      4100      STA HPOSP1
5407 ADE75F      4120      LDA BUGH2
540A 8D02D0      4140      STA HPOSP2
540D ADE85F      4160      LDA BUGH3
5410 8D03D0      4180      STA HPOSP3
                 4200 ;
5413 68          4220      PLA
5414 40          4240      RTI
                 4260 ;
                 4280 ; TOP FLAG DLI
                 4300 ;
5415             4320 TFDLI
5415 48          4340      PHA
5416 A9CE        4360      LDA # <DLI2
5418 8D0002      4380      STA $0200
541B A953        4400      LDA # >DLI2
541D 8D0102      4420      STA $0201
5420 A900        4440      LDA #0       ; set index to FDLISEQ
5422 8DE371      4460      STA FDLINDX
5425 ADE471      4480      LDA FTOPDLI ; need top DLI?
5428 F004        4500      BEQ DLI220
542A 68          4520      PLA          ; yes restore stack
542B 4CA271      4540      JMP FDLI     ; do flag DLI
542E             4560 DLI220
542E 68          4580      PLA
542F 40          4600      RTI
                 4620 ;
                 4640 ;
```

Moore Business Forms, Inc. sv 1412RH

```
                        4640 ;
```



```
                        4660 ; DRAW THE PLAYERS EXTRA CARS
                        4680 ; IN HMEM.  NUMBER OF CARS DRAWN
                        4700 ; IS "CARLFT"
                        4720 ;
5430                    4740 ECAR
5430 A956               4760     LDA # <HMEM+160 ; init ptr to HMEM
5432 85D4               4780     STA R+0
5434 A948               4800     LDA # >HMEM+160
5436 85D5               4820     STA R+1
5438 A200               4840     LDX #0
543A                    4860 ECAR10
543A A000               4880     LDY #0
543C 84DD               4900     STY R+9
543E                    4920 ECAR20
543E A5DD               4940     LDA R+9      ; get count of chars drawn
5440 C905               4960     CMP #5
5442 F023               4980     BEQ ECAR50   ; done with row
5444 CD9642             5000     CMP CARLFT   ; check written enough cars yet
5447 B00F               5020     BCS ECAR30   ; yes - branch to write blanks
5449 BDFA54             5040     LDA ECARDAT,X ; write car data, get byte
544C 91D4               5060     STA (R+0),Y
544E E8                 5080     INX          ; write next byte
544F C8                 5100     INY
5450 BDFA54             5120     LDA ECARDAT,X
5453 91D4               5140     STA (R+0),Y
5455 4C6054             5160     JMP ECAR40
5458                    5180 ECAR30
5458 A900               5200     LDA #0       ; write blanks
545A 91D4               5220     STA (R+0),Y
545C E8                 5240     INX
545D C8                 5260     INY
545E 91D4               5280     STA (R+0),Y
5460                    5300 ECAR40
5460 E6DD               5320     INC R+9      ; inc count of cars drawn on row
5462 CA                 5340     DEX          ; continue - set X back to data for next
                                              car
5463 C8                 5360     INY          ; next space
5464 4C3E54             5380     JMP ECAR20   ; next character
                        5400 ;
5467                    5420 ECAR50
5467 E016               5440     CPX #22      ; check done (12 rows*2 bytes/row  -1 for
                                              old index -1 for starting at 0
5469 B010               5460     BCS ECAR60
546B E8                 5480     INX          ; set to new data
546C E8                 5500     INX
546D A5D4               5520     LDA R+0      ; update HMEM ptr to next row
546F 18                 5540     CLC
5470 6928               5560     ADC #40      ; 40 bytes/line
5472 85D4               5580     STA R+0
5474 90C4               5600     BCC ECAR10
5476 E6D5               5620     INC R+1
5478 4C3A54             5640     JMP ECAR10   ; do next row
547B                    5660 ECAR60
547B 60                 5680     RTS
                        5700 ;
                        5720 ; INIT THE HEADER MEM.
                        5740 ;
547C                    5760 IHMEM
```

```
547C A24F       5780     LDX #79     ; init text from table
547E            5800 IHME10
547E BD1255     5820     LDA IHTXT,X
5481 9DB647     5840     STA HMEM,X
5484 CA         5860     DEX
5485 10F7       5880     BPL IHME10
                5900 ;
                5920 ; clear next header bytes for mode E
                5940 ;
5487 A203       5960     LDX # >HMEMSIZ-80
5489 A000       5980     LDY #0
548B A906       6000     LDA # <HMEM+80
548D 85D4       6020     STA R+0
548F A948       6040     LDA # >HMEM+80
5491 85D5       6060     STA R+1
5493 A900       6080     LDA #0
5495            6100 IHME20
5495 91D4       6120     STA (R+0),Y
5497 C8         6140     INY
5498 D0FB       6160     BNE IHME20
549A CA         6180     DEX
549B F005       6200     BEQ IHME30
549D E6D5       6220     INC R+1     ; next page
549F 4C9554     6240     JMP IHME20
54A2            6260 IHME30
54A2 A220       6280     LDX # <HMEMSIZ-80
54A4            6300 IHME40
54A4 91D4       6320     STA (R+0),Y
54A6 C8         6340     INY
54A7 CA         6360     DEX
54A8 D0FA       6380     BNE IHME40
54AA 60         6400     RTS
                6420 ;
                6440 ; DRAW A FLAG TO A MODE "D" OR
                6460 ; A MODE "E" SCREEN.
                6480 ;
                6500 ; Dest. is in (R+0,R+1)
                6520 ; Color mask is passed in R+3
                6540 ;
                6560 ; R+3=$00     black (invisible)
                6580 ;     $55     white
                6600 ;     $AA     blue
                6620 ;     $FF     green
                6640 ;
54AB            6660 EFLAG
54AB 8A         6680     TXA
54AC 48         6700     PHA
                6720 ;
54AD A200       6740     LDX #0
54AF A000       6760     LDY #0
54B1            6780 EFLA10
54B1 BDDA54     6800     LDA FLAGDAT,X ; get data
54B4 25D7       6820     AND R+3       ; set color
54B6 91D4       6840     STA (R+0),Y
54B8 E8         6860     INX           ; next data
54B9 BDDA54     6880     LDA FLAGDAT,X
54BC 25D7       6900     AND R+3
54BE A001       6920     LDY #1        ; write 2nd column (figure is 2 bytes
```

Moore Business Forms, Inc. sv 1412RH

```
                                        wide)
54C0 91D4       6940    STA (R+0),Y
54C2 E8         6960    INX
54C3 E020       6980    CPX #32         ; done?
54C5 F010       7000    BEQ EFLA20      ; yes - branch
54C7 A000       7020    LDY #0          ; else next line (inc by 2 lines)
54C9 A5D4       7040    LDA R+0
54CB 18         7060    CLC
54CC 6928       7080    ADC #40
54CE 85D4       7100    STA R+0
54D0 90DF       7120    BCC EFLA10
54D2 E6D5       7140    INC R+1
54D4 4CB154     7160    JMP EFLA10
54D7            7180 EFLA20
                7200 ;
54D7 68         7220    PLA
54D8 AA         7240    TAX
54D9 60         7260    RTS
                7280 ;
                7300 ; data for flag
                7320 ;
54DA            7340 FLAGDAT
                7360    ;               FLAG DATA FOR MODE E
54DA 0000       7380    .BYTE $00,$00
54DC 3F00       7400    .BYTE $3F,$00
54DE 3FC0       7420    .BYTE $3F,$C0
54E0 3FF0       7440    .BYTE $3F,$F0
54E2 3FFC       7460    .BYTE $3F,$FC
54E4 3FFC       7480    .BYTE $3F,$FC
54E6 3FF0       7500    .BYTE $3F,$F0
54E8 3FC0       7520    .BYTE $3F,$C0
54EA 3F00       7540    .BYTE $3F,$00
54EC 3000       7560    .BYTE $30,$00
54EE 3000       7580    .BYTE $30,$00
54F0 3000       7600    .BYTE $30,$00
54F2 3000       7620    .BYTE $30,$00
54F4 3000       7640    .BYTE $30,$00
54F6 3000       7660    .BYTE $30,$00
54F8 0000       7680    .BYTE $00,$00
                7700    ;               END OF FLAG DATA
                7720 ;
                7740 ; data for player's extra cars
                7760 ; (12 rows x 2 columns)
54FA            7780 ECARDAT
                7800 ;          .BYTE 242,143,250,175,234,171
                7820 ;  .BYTE 42,168,32,8,32,8,32,8,32,8,34,136,234,171,250,17
                                      5,240,15
54FA 00000690   7840    .BYTE 0,0,6,144,58,172,58,172,57,108,5,80
54FE 3AAC3AAC
5502 396C0550
5506 05500550   7860    .BYTE 5,80,5,80,58,172,58,172,62,188,60,60
550A 3AAC3AAC
550E 3EBC3C3C
                7880 ;
                7900 ;
                7920 ; table to init HMEM with chars
                7940 ;
5512            7960 IHTXT
```

```
5512 100F1312  7980     .BYTE 16,15,19,18,1,24,24,24,24,24,24,0,24
5516 01181818
551A 18181800
551E 18
551F 1C1D1E1F  8000     .BYTE 28,29,30,31,32,33,34,35,36,37,38,39,40,41
5523 20212223
5527 24252627
552B 2829
552D 180D0E0C  8020     .BYTE 24,13,14,12,13,24,24,24,24,24,24,24,0
5531 0D181818
5535 18181818
5539 00
553A 100F1312  8040     .BYTE 16,15,19,18,2,24,24,24,24,24,24,0,24
553E 02181818
5542 18181800
5546 18
5547 2A2B2C2D  8060     .BYTE 42,43,44,45,46,47,48,49,50,51,52,53,54,55
554B 2E2F3031
554F 32333435
5553 3637
5555 18121415  8080     .BYTE 24,18,20,21,22,23,24,24,24,24,24,24,0
5559 16171818
555D 18181818
5561 00
               8100 ;
               8120 ;
               8140 ; INIT TEXT/TITLE CHARS
               8160 ; OS chars needed for text are copied
               8180 ; to TCHAR mem. and CHBASE is set
               8200 ; to point to TCHAR
               8220 ;
5562           8240 ITCHARS
5562 A900      8260     LDA # <TCHARS ; init dest. ptr. to TCHAR area
5564 85D6      8280     STA R+2
5566 A958      8300     LDA # >TCHARS
5568 85D7      8320     STA R+3
556A A200      8340     LDX #0       ; init index to char#'s to be copied
556C           8360 ITCH10
556C BDA555    8380     LDA CHCOPY,X ; get char# to copy
556F 302E      8400     BMI ITCH30   ; 0 is end of data
5571 A000      8420     LDY #0
5573 84D5      8440     STY R+1
5575 0A        8460     ASL A        ; 8 bytes/char
5576 26D5      8480     ROL R+1
5578 0A        8500     ASL A
5579 26D5      8520     ROL R+1
557B 0A        8540     ASL A
557C 26D5      8560     ROL R+1
557E 85D4      8580     STA R+0
5580 A5D5      8600     LDA R+1      ; add in offset to OS char base
5582 18        8620     CLC
5583 69E0      8640     ADC # >OSCHARS
5585 85D5      8660     STA R+1
               8680 ;
5587           8700 ITCH15
5587 A007      8720     LDY #7
5589           8740 ITCH20
5589 B1D4      8760     LDA (R+0),Y
```

Moore Business Forms, Inc. sv 1412RH

```
558B 91D6      8780     STA (R+2),Y
558D 88        8800     DEY
558E 10F9      8820     BPL ITCH20
               8840 ;
5590 E8        8860     INX          ; next char
5591 A5D6      8880     LDA R+2      ; inc dest ptr
5593 18        8900     CLC
5594 6908      8920     ADC #8
5596 85D6      8940     STA R+2
5598 90D2      8960     BCC ITCH10
559A E6D7      8980     INC R+3
559C 4C8C55    9000     JMP ITCH10
559F           9020 ITCH30
559F A958      9040     LDA # >TCHARS ; set CHBAS to new charset
55A1 8DF402    9060     STA CHBAS
55A4 60        9080     RTS
               9100 ; CHCOPY is a list of char#'s to
               9120 ; be copied from the OS charset
               9140 ;
               9160 ; chars copied are:
55A5           9180 CHCOPY
               9200 ;          0, 1, 2, 3, 4, 5, 6, 7
55A5 10111213  9220     .BYTE 16,17,18,19,20,21,22,23
55A9 14151617
               9240 ;          8, 9, A, E, G, H, I, L
55AD 18192125  9260     .BYTE 24,25,33,37,39,40,41,44
55B1 2728292C
               9280 ;          P, Q, R, Y, O, U, N, D
55B5 30313239  9300     .BYTE 48,49,50,57,47,53,46,36
55B9 2F352E24
               9320 ;          space, T, I, :, end
55BD 00342D1A  9340     .BYTE 0,52,45,26,128
55C1 80
               9360 ;
               9380 ; Reserve space for TCHAR
               9400 ; (title and text) chars
               9420 ;  must align to 1K
55C2           9440     .DS AA*+1023U&$FC00U-*
5800           9460 TCHARS
5800           9480     .DS 28*8   ; reserve for OS char copy
               9500 ; UPN DOWN title chars are at #28-55
58E0 0F1F1E1E  9520     .BYTE 15,31,30,30,62,60,124,120
58E4 3E3C7C78
58E8 0F0F0F1F  9540     .BYTE 15,15,15,31,30,30,60,61
58EC 1E1E3C3D
58F0 8080000E  9560     .BYTE 128,128,0,14,15,30,28,61
58F4 0F1E1C3D
58F8 000000E1  9580     .BYTE 0,0,0,225,243,115,231,231
58FC F373E7E7
5900 000000DF  9600     .BYTE 0,0,0,223,183,231,207,143
5904 B7E7CF8F
5908 00000000  9620     .BYTE 0,0,0,0,128,128,0,0
590C 80800000
5910 0F1F3F00  9640     .BYTE 15,31,63,0,30,62,124,120
5914 1E3E7C78
5918 FCFFFF07  9660     .BYTE 252,255,255,7,15,15,15,31
591C 0F0F0F1F
5920 00008083  9680     .BYTE 0,0,128,131,143,30,60,56
```

Moore Business Forms, Inc. sv 1412RH

```
                                5924 8F1E3C38
5928 000000F0   9700    .BYTE 0,0,0,240,249,57,123,243
592C F9397BF3
5930 000000F3   9720    .BYTE 0,0,0,243,227,199,207,159
5934 E3C7CF9F
5938 000000C7   9740    .BYTE 0,0,0,199,199,143,30,28
593C C78F1E1C
5940 0000008E   9760    .BYTE 0,0,0,142,157,31,62,60
5944 9D1F3E3C
5948 000000F8   9780    .BYTE 0,0,0,248,188,60,56,120
594C BC3C3878
5950 783F3F00   9800    .BYTE 120,63,63,0,0,0,0,0
5954 00000000
5958 FFFE7806   9820    .BYTE 255,254,120,6,31,63,31,15
595C 1F3F1F0F
5960 397F77E0   9840    .BYTE 57,127,119,224,192,192,128,0
5964 C0C08000
5968 CF8E9E00   9860    .BYTE 207,142,158,0,0,0,0,0
596C 00000000
5970 0E1E1C3D   9880    .BYTE 14,30,28,61,63,63,63,30
5974 3F3F3F1E
5978 00000180   9900    .BYTE 0,0,1,128,192,192,128,0
597C C0C08000
5980 F8FFFF00   9920    .BYTE 248,255,255,0,0,0,0,0
5984 00000000
5988 7FFCF800   9940    .BYTE 127,252,248,0,0,0,0,0
598C 00000000
5990 3D1F0F00   9960    .BYTE 61,31,15,0,0,0,0,0
5994 00000000
5998 F7C78700   9980    .BYTE 247,199,135,0,0,0,0,0
599C 00000000
59A0 9EFF1F00   010000  .BYTE 158,255,31,0,0,0,0,0
59A4 00000000
59A8 3CFCBE00   010020  .BYTE 60,252,190,0,0,0,0,0
59AC 00000000
59B0 7870F001   010040  .BYTE 120,112,240,1,1,1,1,0
59B4 01010100
59B8 70F0E0EC   010060  .BYTE 112,240,224,236,254,254,252,240
59BC FEFEFCF0
                015720  .INCLUDE #D2:DRAW.M65
                1000    .PAGE "DRAW/ERASE"
```

Moore Business Forms, Inc. sv 1412BH

```
                    1160 ;
                    1180 ; DRAW ROUTINE - DRAWS OBJECT
                    1200 ; IN OBJTBL INDEXED BY Xreg
                    1220 ;
 59C0               1240 DRAW
 59C0 86EC          1260      STX R+24    ; preserve Xreg
                    1290 ;
 59C2               1292 DRAW03
 59C2 A901          1300      LDA #1      ; flag draw setup
 59C4 20525A        1320      JSR DESETUP ; setup pointers and variables for draw
 59C7 B03D          1340      BCS DRAW30  ; branch if not displayable
                    1660 ;
 59C9               1680 DRAW04
 59C9 A910          1700      LDA #16     ; # rows to draw
 59CB 85E7          1720      STA R+19
 59CD A000          1740      LDY #0
 59CF               1760 DRAW05
 59CF A205          1780      LDX #5      ; #bytes/row
 59D1               1800 DRAW10
 59D1 B1DE          1820      LDA (R+10),Y ; get current background
 59D3 91E0          1840      STA (R+12),Y ; save
 59D5 31E2          1860      AND (R+14),Y ; create "hole" with mask
 59D7 11E4          1880      ORA (R+16),Y ; "OR" new data
 59D9 91DE          1900      STA (R+10),Y ; write to screen
 59DB C8            1920      INY
 59DC CA            1940      DEX         ; dec count of bytes in row
 59DD D0F2          1960      BNE DRAW10
                    1980 ;
 59DF C6E7          2000      DEC R+19    ; dec count of rows to draw
 59E1 F022          2020      BEQ DRAW20  ; branch when done
                    2040 ;
 59E3 C6EA          2060      DEC R+22    ; time to wrap?
 59E5 D010          2080      BNE DRAW15  ; no - branch
 59E7 A5DE          2100      LDA R+10    ; else reset ptr
 59E9 38            2120      SEC         ; to buff beginning
 59EA E95D          2140      SBC # <A112*40-35
 59EC 85DE          2160      STA R+10
 59EE A5DF          2180      LDA R+11
 59F0 E911          2200      SBC # >A112*40-35
 59F2 85DF          2220      STA R+11
 59F4 4CCF59        2240      JMP DRAW05
 59F7               2260 DRAW15
 59F7 A5DE          2280      LDA R+10
 59F9 18            2300      CLC
 59FA 6923          2320      ADC #35     ; inc ptr by 40 bytes/line - Yreg inc.
 59FC 85DE          2340      STA R+10
 59FE 90CF          2360      BCC DRAW05
 5A00 E6DF          2380      INC R+11
 5A02 4CCF59        2400      JMP DRAW05
 5A05               2420 DRAW20
 5A05 18            2440      CLC         ; flag successful
 5A06               2460 DRAW30
 5A06 A6EC          2480      LDX R+24
 5A08 60            2520      RTS
                    2540 ;
                    2560 ;
                    2580 ; ERASE ROUTINE - ERASE ENTRY IN
                    2600 ; OBJTBL INDEXED BY Xreg
```

```
5A08 AAEC    2480     LDX R+24
5A08 60      2520     RTS
             2540 ;
             2560 ;
             2580 ; ERASE ROUTINE - ERASE ENTRY IN
             2600 ; OBJTBL INDEXED BY Xreg
```



```
             2620 ;
5A09         2640 ERASE
5A09 8A      2660     TXA
5A0A 48      2680     PHA
5A0B A900    2700     LDA #0      ; flag erase setup
5A0D 20525A  2720     JSR DESETUP ; setup for erase
5A10 B03D    2740     BCS ERAS30  ; jump if cant erase
             2760 ;
5A12 A96E    2820     LDA #110
5A14         2830 ERAS06
5A14 CD0BD4  2840     CMP VCOUNT  ; sync to screen
5A17 90FB    2850     BCC ERAS06
             2860 ;
             2980 ;
5A19         3000 ERAS04
5A19 A000    3020     LDY #0
5A1B A910    3040     LDA #16     ; # rows to draw
5A1D 85E7    3060     STA R+19
5A1F         3080 ERAS05
5A1F A205    3100     LDX #5
5A21         3120 ERAS10
5A21 B1E0    3140     LDA (R+12),Y ; get saved background
5A23 91DE    3160     STA (R+10),Y ; restore to screen
5A25 C8      3180     INY
5A26 CA      3200     DEX
5A27 D0F8    3220     BNE ERAS10
             3240 ;
5A29 C6E7    3260     DEC R+19
5A2B F022    3280     BEQ ERAS30
             3300 ;
5A2D C6EA    3320     DEC R+22    ; time to wrap?
5A2F D010    3340     BNE ERAS15
5A31 A5DE    3360     LDA R+10
5A33 38      3380     SEC
5A34 E95D    3400     SBC # <A112*40-35
5A36 85DE    3420     STA R+10
5A38 A5DF    3440     LDA R+11
5A3A E911    3460     SBC # >A112*40-35
5A3C 85DF    3480     STA R+11
5A3E 4C1F5A  3500     JMP ERAS05
5A41         3520 ERAS15
5A41 A5DE    3540     LDA R+10
5A43 18      3560     CLC
5A44 6923    3580     ADC #35
5A46 85DE    3600     STA R+10
5A48 90D5    3620     BCC ERAS05
5A4A E6DF    3640     INC R+11
5A4C 4C1F5A  3660     JMP ERAS05
5A4F         3680 ERAS30
5A4F 68      3700     PLA
5A50 AA      3720     TAX
5A51 60      3780     RTS
             3820 ;
             3840 ; DRAW/ERASE SETUP
             3860 ; SET REGISTERS WITH INFORMATION
             3880 ; NEEDED TO DO DRAW OR ERASE
             3900 ; IF Accum. = 0, ERASE IS SETUP
```

Moore Business Forms, Inc. sv 1412RH

```
3880 ; NEEDED TO DO DRAW OR ERASE
3900 ; IF Accum. = 0, ERASE IS SETUP
3920 ; IF Accum.<> 0, DRAW IS SETUP
```



```
                    3940 ;
                    3960 ; R+10,11      PTR TO SCREEN
                    3980 ; R+12,13      PTR TO BACKSAVE AREA
                    4000 ; R+14,15      PTR TO MASK
                    4020 ; R+16,17      PTR TO DATA
                    4040 ; R+18         INIT. SCREEN ROW
                    4060 ; R+20,21      USED
                    4080 ; R+22         WRAPAROUND COUNT -
                    4100 ;              # lines from end of
                    4120 ;              buffer current ptr is
                    4140 ; USES R+23
                    4160 ;
5A52                4180 DESETUP
5A52 48             4200    PHA         ; save flag for draw or erase setup
                    4220 ;
                    4240 ; Displayable?
                    4260 ;
5A53 BDFE62         4280    LDA YPOS,X  ; Y range is 0-95
5A56 C960           4300    CMP #96
5A58 9003           4320    BCC DESE10
5A5A 68             4340    PLA         ; remove flag from stack
5A5B 38             4360    SEC         ; flag not successful
5A5C 60             4380    RTS
                    4400 ;
                    4420 ; Set wraparound counter
                    4440 ;
5A5D                4460 DESE10
5A5D 18             4480    CLC
5A5E 6D0A40         4500    ADC DSCROL  ; Y is relative to top of screen
5A61 A8             4520    TAY
5A62 85EA           4540    STA R+22
5A64 A970           4560    LDA #112
5A66 38             4580    SEC
5A67 E5EA           4600    SBC R+22
5A69 B002           4620    BCS DESE15
5A6B 6970           4640    ADC #112
5A6D                4660 DESE15
5A6D 85EA           4680    STA R+22
                    4700 ;
                    4720 ; Set initial screen pointer
                    4740 ;
5A6F B93E40         4760    LDA DSPTBL,Y
5A72 85DE           4780    STA R+10
5A74 B91D41         4800    LDA DSPTBH,Y
5A77 85DF           4820    STA R+11
5A79 BDF562         4840    LDA XPOS,X  ; get X
5A7C 4A             4860    LSR A       ; convert to byte#
5A7D 4A             4880    LSR A
5A7E 18             4900    CLC
5A7F 65DE           4920    ADC R+10    ; add to pointer
5A81 85DE           4940    STA R+10
5A83 9002           4960    BCC DESE20
5A85 E6DF           4980    INC R+11
                    5000 ;
                    5020 ; Init backsave ptr
                    5040 ;
5A87                5060 DESE20
```

Moore Business Forms, Inc. sv 1412RH

```
5A80    TXA             ; OBJTBL index determines backsave area
```



```
5A88 A8         5100    TAY
5A89 B9136D     5120    LDA BSTBLL,Y ; address low
5A8C 85E0       5140    STA R+12
5A8E B91B6D     5160    LDA BSTBLH,Y ;    " high
5A91 85E1       5180    STA R+13
5A93 68         5200    PLA             ; check draw or erase setup
5A94 F03F       5220    BEQ DESE30  ; if erase then done
                5240 ;
                5260 ; Init ptr to mask
                5280 ;
5A96 BDF562     5300    LDA XPOS,X  ; form index to data/mask tables
5A99 2903       5320    AND #$03    ; XPOS determines shift
5A9B 85EB       5340    STA R+23
5A9D BD0763     5360    LDA DIR,X   ; dir determines image
5AA0 C902       5380    CMP #2      ; check for truck undef dir.
5AA2 9010       5400    BCC DESE57
5AA4 F00C       5420    BEQ DESE56
5AA6 38         5440    SEC
5AA7 E901       5460    SBC #1
5AA9 C904       5480    CMP #4
5AAB 9007       5500    BCC DESE57
5AAD A902       5520    LDA #2
5AAF 4CB45A     5540    JMP DESE57
5AB2            5560 DESE56
5AB2 A900       5580    LDA #0
5AB4            5600 DESE57
5AB4 0A         5620    ASL A
5AB5 0A         5640    ASL A
5AB6 18         5660    CLC
5AB7 65EB       5680    ADC R+23
5AB9 0A         5700    ASL A
5ABA BCEC62     5720    LDY TYPE,X
5ABD 795A6B     5740    ADC OBJREF,Y ; carry still clear
5AC0 A8         5760    TAY
5AC1 B9C56B     5780    LDA OBJMSK,Y
5AC4 85E2       5800    STA R+14
5AC6 B9C66B     5820    LDA OBJMSK+1,Y
5AC9 85E3       5840    STA R+15
                5860 ;
                5880 ; Init ptr to data
                5900 ;
5ACB B9856B     5920    LDA OBJDAT,Y ; data addr. high
5ACE 85E4       5940    STA R+16
5AD0 B9866B     5960    LDA OBJDAT+1,Y ;   "  "  low
5AD3 85E5       5980    STA R+17
                6000 ;
5AD5            6020 DESE30
5AD5 18         6040    CLC             ; flag successful
5AD6 60         6060    RTS
                015740  .INCLUDE #D1:TRACK3.M65
                1000    .PAGE "TRACK"
```

Moore Business Forms, Inc. sv 14128H

```
                    1020 ; TRACK3.M65
                    1040 ;
                    1060 YTRKPT = 11      ; Y track point relative to top of object
   =000B
                    1080 BLSTINC *= *+1
5AD7                1100 BLSTNZR *= *+1
5AD8                1120 BUGDIR *= *+1    ; player's direction
5AD9                1140 BPTHCT *= *+1    ; index to path player is following
5ADA                1160 SCRFLG *= *+1    ; flag set when scrolling occurs
5ADB                1180 JMPDLY = 12      ; delay during jump sequencing
   =000C
5ADC                1200 JMPCTR *= *+1    ; count for delay between jump moves
5ADD                1220 OLDBUGY *= *+1   ; player's last Y position
5ADE                1240 OLDBDIR *= *+1   ; player's last direction
   =D300            1260 PORTA = $D300     ; joystick port
   =D302            1280 PACTL = $D302     ; PORTA control
   =0010            1300 FWDSPD = 16       ; max fwd. speed (must be positive)
   =FFF0            1320 BWDSPD = -16      ; max bwd. speed (must be negative)
   =0028            1340 ACCDLY = 40       ; delay betw. accel. incs.
5ADF                1360 ACCCNT *= *+1     ; counter for accel delay
5AE0                1380 JMPNDX *= *+1     ; index to jump path
   =D010            1400 TRIGO = $D010     ; joystick trigger
   =0008            1420 NPATH = 8         ; # path iterations to get across
                                           intersection
5AE1                1440 WHLSEQ *= *+1     ; sequence count for wheel spin
                    1460 ;
                    1480 ;
                    1500 ; ROUTINE RETURNS X FOR A VEHICLE
                    1520 ; SUCH THAT VEHICLE TRACKS THE ROAD
                    1540 ;
                    1560 ; ENTER WITH:
                    1580 ;       R+0     present X (0-159)
                    1600 ;       R+1     new Y (0-95)
                    1620 ;       R+2     present direction
                    1640 ;
                    1660 ;               left    right
                    1680 ;               0   level   3
                    1700 ;               1   up      4
                    1720 ;               2   down    5
                    1740 ;       R+26    last increment
                    1760 ;       R+27    last non-zero incr.
                    1780 ;       R+28    speed (negative if in reverse)
                    1800 ;       R+8     forced path counter
                    1820 ; RETURNS:
                    1840 ;       R+0     new X
                    1860 ;       R+3     road status byte
                    1880 ;               0-5  direction
                    1900 ;               128 = hit ground
                    1920 ;               129 = hit water
                    1940 ;               bit6=1, turning point
                    1960 ; USES:
                    1980 ;       R+4,5,6,7,9
                    2000 TRACK
5AE2                2020        TYA
5AE2 98             2040        PHA
5AE3 48             2060        TXA
5AE4 8A             2080        PHA
5AE5 48             2100 TRAC02
5AE6                2120        LDA R+8     ; on path or tracking?
5AE6 A5DC           2140        BEQ TRAC10  ; tracking - branch
5A58 F029           4400        FOR #
5B7D 49FF           4400        EOR #
5B7F 18
```

```
5AE8 F029       2140      BEQ TRAC10  ; tracking - branch
```



```
                2160 ;
                2180 ; On path - get next byte and add
                2200 ; as Xpos. increment
                2220 ;
5AEA 2903       2240      AND #3       ; calculate path index
5AEC 85DD       2260      STA R+9
5AEE A5D6       2280      LDA R+2
5AF0 0A         2300      ASL A
5AF1 0A         2320      ASL A
5AF2 05DD       2340      ORA R+9
5AF4 AA         2360      TAX          ; path index in Xreg
5AF5 BD245C     2380      LDA PATHD,X  ; get inc.
5AF8 A6F0       2400      LDX R+28     ; forward or backward?
5AFA 1005       2420      BPL TRAC05
5AFC 49FF       2440      EOR #$FF     ; invert addend
5AFE 18         2460      CLC
5AFF 6901       2480      ADC #1
5B01            2500 TRAC05
5B01 85EE       2520      STA R+26     ; save as last inc.
5B03 0900       2540      ORA #0       ; set zero flag
5B05 F002       2560      BEQ TRAC06
5B07 85EF       2580      STA R+27     ; save as last non-zero
5B09            2600 TRAC06
5B09 18         2620      CLC          ; add to Xpos.
5B0A 65D4       2640      ADC R+0
5B0C 85D4       2660      STA R+0
                2680 ;
5B0E C6DC       2700      DEC R+8      ; update index
5B10 4CA95B     2720      JMP TRAC70
                2740 ;
                2760 ; Track the left edge of the road
                2780 ;
5B13            2800 TRAC10
5B13 A5D5       2820      LDA R+1      compute ptr to screen from X,Y
5B15 18         2840      CLC
5B16 6D0A40     2860      ADC DSCROL
5B19 690B       2880      ADC #YTRKPT
5B1B A8         2900      TAY
5B1C A5D4       2920      LDA R+0      ; get x
5B1E 4A         2940      LSR A        ; find horizontal byte offset
5B1F 4A         2960      LSR A
5B20 18         2980      CLC
5B21 793E40     3000      ADC DSPTBL,Y ; low byte
                3020 ;
5B24 85DA       3040      STA R+6
5B26 B91D41     3060      LDA DSPTBH,Y ; high byte
5B29 6900       3080      ADC #0
5B2B 85DB       3100      STA R+7
                3120 ;
5B2D A5DA       3140      LDA R+6      ; -5 to allow search of left byte
5B2F 38         3160      SEC
5B30 E905       3180      SBC #5
5B32 85DA       3200      STA R+6
5B34 B002       3220      BCS TRAC18
5B36 C6DB       3240      DEC R+7
                3260 ;
5B38            3280 TRAC18
5B38 A202       3300      LDX #2       ; # bytes to search-1
```

Moore Business Forms, Inc. sv 1412RH

```
5B38                    3280 TRAC18
5B38 A202               3300     LDX #2      ; # bytes to search-1
```



```
5B3A A006               3320     LDY #6      ; rightmost byte to search (middle is #5)
5B3C                    3340 TRAC20
5B3C B1DA               3360     LDA (R+6),Y ; find a non-road byte
5B3E C9FF               3380     CMP #$FF
5B40 D00B               3400     BNE TRAC30
                        3420 ;   LDA #$1B
                        3440 ;   STA (R+6),Y
5B42 88                 3460     DEY
5B43 CA                 3480     DEX
5B44 10F6               3500     BPL TRAC20
                        3520 ;
                        3540 ; search failed
                        3560 ;
5B46 A908               3580     LDA #NPATH  ; set on path according to current
                                 direction
5B48 85DC               3600     STA R+8
5B4A 4CE65A             3620     JMP TRAC02
                        3640 ;
5B4D                    3660 TRAC30
5B4D 48                 3680     PHA         ; save left road byte
5B4E A5D4               3700     LDA R+0
5B50 29FC               3720     AND #$FC
5B52 18                 3740     CLC
5B53 7D215C             3760     ADC BYTADD,X ; add byte offset to new X
5B56 85DD               3780     STA R+9
                        3800 ;
5B58 68                 3820     PLA
5B59 A203               3840     LDX #3
5B5B                    3860 TRAC40
5B5B 6A                 3880     ROR A       ; find pixel offset of L side of road
5B5C 9006               3900     BCC TRAC50
5B5E 6A                 3920     ROR A
5B5F 9003               3940     BCC TRAC50
5B61 CA                 3960     DEX
5B62 D0F7               3980     BNE TRAC40
                        4000 ;
5B64                    4020 TRAC50
                        4040 ;   ORA BB,X
5B64 8A                 4060     TXA
5B65 18                 4080     CLC
5B66 65DD               4100     ADC R+9
5B68 85DD               4120     STA R+9
5B6A 38                 4140     SEC
5B6B E5D4               4160     SBC R+0
5B6D 85D9               4180     STA R+5     ; save increment
5B6F A5DD               4200     LDA R+9     ; set new X
5B71 85D4               4220     STA R+0
5B73 A6D4               4240     LDX R+0
5B75 A4D5               4260     LDY R+1
                        4280 ;
                        4300 ; set new direction
                        4320 ;
5B77 A5D9               4340     LDA R+5     ; get inc
5B79 A6F0               4360     LDX R+28    ; fwd or bwd?
5B7B 1007               4380     BPL TRAC48
5B7D 49FF               4400     EOR #$FF
5B7F 18                 4420     CLC
```

Moore Business Forms, Inc. 5V 1412M

```
5B80 6901       4440     ADC #1

PAGE 38   UPN DOWN - for Atari systems
TRACK                                                D1:TRACK3.M65

5B82 85D9       4460     STA R+5
5B84            4480 TRAC48
5B84 AA         4500     TAX
5B85 E4EE       4520     CPX R+26    ; cmp to last inc
5B87 D008       4540     BNE TRAC56
5B89 A5EE       4560     LDA R+26
5B8B F012       4580     BEQ TRAC60
5B8D E8         4600     INX
5B8E E8         4620     INX
5B8F BD1B5C     4640     LDA DIRINC,X
5B92 85D6       4660     STA R+2
5B94            4680 TRAC56
5B94 A5D9       4700     LDA R+5
5B96 85EE       4720     STA R+26    ; save as last inc
5B98 F00F       4740     BEQ TRAC65
5B9A 85EF       4760     STA R+27    ; save as last non-zero
5B9C 4CA95B     4780     JMP TRAC65
5B9F            4800 TRAC60
5B9F A204       4820     LDX #4
5BA1 A5EF       4840     LDA R+27
5BA3 1002       4860     BPL TRAC64
5BA5 A201       4880     LDX #1
5BA7            4900 TRAC64
5BA7 86D6       4920     STX R+2
5BA9            4940 TRAC65
                4960 ;
                4980 ; Calculate pointer to status byte
                5000 ;
5BA9            5020 TRAC70
5BA9 A5D4       5040     LDA R+0     ; ptr=SCROLC+A(Y-32)/320*10+(X+4)/16
5BAB 18         5060     CLC         ; offset X
5BAC 6908       5080     ADC #8
5BAE 4A         5100     LSR A
5BAF 4A         5120     LSR A
5BB0 4A         5140     LSR A
5BB1 4A         5160     LSR A
5BB2 85D8       5180     STA R+4
5BB4 A5D5       5200     LDA R+1     ; get Y
5BB6 18         5220     CLC
5BB7 690B       5240     ADC #YTRKPT
5BB9 18         5260     CLC
5BBA 6D0D40     5280     ADC SCROLF
5BBD 4A         5300     LSR A
5BBE 4A         5320     LSR A
5BBF 4A         5340     LSR A
5BC0 4A         5360     LSR A
5BC1 A8         5380     TAY
5BC2 B9264B     5400     LDA M10TBL,Y ; *10
5BC5 18         5420     CLC
5BC6 65D8       5440     ADC R+4     ; won't ovfl
5BC8 65CA       5460     ADC SCROLC
5BCA 85D8       5480     STA R+4
5BCC A900       5500     LDA #0
5BCE 65CB       5520     ADC SCROLC+1
5BD0 85D9       5540     STA R+5
                5560 ;
                5580 ; Correct for overflow
                5600 ;
```

Moore Business Forms, Inc. sv 1412RM

```
5BCE 65CB       5520      ADC SCROLC+1
5BD0 85D9       5540      STA R+5
                5560 ;
                5580 ; Correct for overflow
                5600 ;
```



```
5BD2 A5D9       5620      LDA R+5
5BD4 C946       5640      CMP # >APFDATA+PFLENU
5BD6 9015       5660      BCC TRAC90
5BD8 D006       5680      BNE TRAC80
5BDA A5D8       5700      LDA R+4
5BDC C922       5720      CMP # <APFDATA+PFLENU
5BDE 900D       5740      BCC TRAC90
5BE0            5760 TRAC80
5BE0 A5D8       5780      LDA R+4
5BE2 38         5800      SEC
5BE3 E96C       5820      SBC # <PFLEN
5BE5 85D8       5840      STA R+4
5BE7 A5D9       5860      LDA R+5
5BE9 E902       5880      SBC # >PFLEN
5BEB 85D9       5900      STA R+5
                5920 ;
                5940 ; Get road status byte
                5960 ;
5BED            5980 TRAC90
5BED A000       6000      LDY #0
5BEF B1D8       6020      LDA (R+4),Y ; get char #
5BF1 AA         6040      TAX
5BF2 BD5E5E     6060      LDA RDSTAT1,X ; get byte from table
5BF5 85D7       6080      STA R+3
                6100 ;
                6120 ; Turn point?
                6140 ;
5BF7 2940       6160      AND #$40    ; check for intrsctn.
5BF9 F01B       6180      BEQ TRA100
5BFB A5D5       6200      LDA R+1
5BFD 18         6220      CLC
5BFE 6D0D40     6240      ADC SCROLF
5C01 690B       6260      ADC #YTRKPT ; won't ovfl
5C03 290F       6280      AND #$0F
5C05 85D9       6300      STA R+5
5C07 BD8C5E     6320      LDA RDSTAT2,X ;check which line turning pt. strobes
5C0A 290F       6340      AND #$0F
5C0C C5D9       6360      CMP R+5
                6380 ;
5C0E F006       6400      BEQ TRA100  ; yes - branch
5C10 A5D7       6420      LDA R+3     ; else no turns
5C12 29BF       6440      AND #$BF
5C14 85D7       6460      STA R+3
5C16            6480 TRA100
5C16 68         6500      PLA
5C17 AA         6520      TAX
5C18 68         6540      PLA
5C19 A8         6560      TAY
5C1A 60         6580      RTS
5C1B 02000103   6600 DIRINC .BYTE 2,0,1,3,5,0
5C1F 0500
                6620 ;
                6640 ; Values to add to Xpos. to
                6660 ; move 1 byte  left, nowhere, right
                6680 ;
5C21 FC0004     6700 BYTADD .BYTE -4,0,4
                6720 ;
```

```
                        6740 ; Paths for "lost" vehicles are
                        6760 ; 4 bytes long and wrap around
                        6780 ;
5C24                    6800 PATHD
5C24 FFFFFFFF           6820     .BYTE -1,-1,-1,-1 ; dir. =0
5C28 FF000000           6840     .BYTE -1,0,0,0 ;     1
5C2C FEFEFEFE           6860     .BYTE -2,-2,-2,-2 ;     2
5C30 01010101           6880     .BYTE 1,1,1,1 ;     3
5C34 01000000           6900     .BYTE 1,0,0,0 ;     4
5C38 02020202           6920     .BYTE 2,2,2,2 ;     5
                        6940 ;
                        6960 ; Jump path
                        6980 ;
5C3C                    7000 JMPADD ;       used to calc. Y during jump
5C3C FEFCFAF8           7020     .BYTE -2,-4,-6,-8,-9,-11,-12,-14,-16,-17
5C40 F7F5F4F2
5C44 F0EF
5C46 EEEDECEB           7040     .BYTE -18,-19,-20,-21,-21,-22,-23,-23,-24,-24
5C4A EBEAE9E9
5C4E E8E8
5C50 E8E8E8E8           7060     .BYTE -24,-24,-24,-24,-24,-24,-24,-23,-23,-22,-21,-21
5C54 E8E8E8E9
5C58 E9EAEBEB
5C5C ECEDEEEF           7080     .BYTE -20,-19,-18,-17,-16,-14,-12,-11,-9,-8,-6,-4,-2
5C60 F0F2F4F5
5C64 F7F8FAFC
5C68 FE
5C69                    7100 JMPPIC ;       used to calc pic during jump
5C69 01010101           7120     .BYTE 1,1,1,1,1,1,1,1,1,1
5C6D 01010101
5671 0101
5C73 01010101           7140     .BYTE 1,1,1,1,1,1,1,0,0,0
5C77 01010100
5C7B 0000
5C7D 00000000           7160     .BYTE 0,0,0,0,0,0,0,0,2,2,2,2
5C81 00000000
5C85 02020202
5C89 02020202           7180     .BYTE 2,2,2,2,2,2,2,2,2,2,2,0,0,0
5C8D 02020202
5C91 02020200
5C95 0000
    =002D               7200 ENDJMP = JMPPIC-JMPADD ; # bytes in each of the above
                                     lists
                        7220 ;
                        7240 ;
                        7260 ; MOVE THE PLAYERS 'BUG'
                        7280 ;
5C97                    7300 PMOVE
5C97 ADA842             7320     LDA BUGX   ; player's X
5C9A 85D4               7340     STA R+0
5C9C ADA942             7360     LDA BUGY   ;     Y
5C9F 85D5               7380     STA R+1
5CA1 ADD75A             7400     LDA BLSTINC
5CA4 85EE               7420     STA R+26
5CA6 ADD85A             7440     LDA BLSTNZR
5CA9 85EF               7460     STA R+27
5CAB ADD95A             7480     LDA BUGDIR  ;     direction
5CAE 85D6               7500     STA R+2
5CB0 ADDC40             7520     LDA SCRSPD
```

```
5CB0 AD0C40     7520     LDA SCRSPD
```



```
5CB3 85F0       7540     STA R+28
                7560 ;
5CB5 ADDA5A     7580     LDA BPTHCT  ;   path count
5CB8 85DC       7600     STA R+8
                7620 ;
                7640 ; Acceleration/deceleration
                7660 ;
5CBA            7680 PMOV10
5CBA CEDF5A     7700     DEC ACCCNT
5CBD D028       7720     BNE PMOV30
5CBF A928       7740     LDA #ACCDLY
5CC1 8DDF5A     7760     STA ACCCNT
5CC4 AD00D3     7780     LDA PORTA   ; check joystick
5CC7 2903       7800     AND #3      ;... for forward/backward pos.
5CC9 C903       7820     CMP #3
5CCB F01A       7840     BEQ PMOV30  ; neither selected
5CCD 2901       7860     AND #1      ; stick forward?
5CCF D00B       7880     BNE PMOV20  ; no - branch
5CD1 AE0C40     7900     LDX SCRSPD  ; accelerate
5CD4 E8         7920     INX
                7940 ;   CPX #128    ; can't go negative
5CD5 E010       7960     CPX #FWDSPD
5CD7 F00E       7980     BEQ PMOV30
5CD9 4CE45C     8000     JMP PMOV25
5CDC            8020 PMOV20
5CDC AE0C40     8040     LDX SCRSPD  ;decelerate
5CDF CA         8060     DEX
                8080 ;   CPX #127    ; can't go positive
5CE0 E0F0       8100     CPX #BWDSPD
5CE2 F003       8120     BEQ PMOV30
5CE4            8140 PMOV25
5CE4 8E0C40     8160     STX SCRSPD
                8180 ;
                8200 ; JUMPS
                8220 ;
5CE7            8240 PMOV30
5CE7 AEE05A     8260     LDX JMPNDX  ; jump in progress?
5CEA 101B       8280     BPL PMOV33  ; yes - branch
                8300 ;
5CEC AD10D0     8320     LDA TRIG0   ; check trigger button
5CEF F003       8340     BEQ PMOV32  ; branch if pressed
5CF1 4CA95D     8360     JMP PMOV60
5CF4            8380 PMOV32
5CF4 AD0C40     8400     LDA SCRSPD  ; can only jump if not moving backwards
5CF7 1003       8420     BPL PMOV31
5CF9 4CA95D     8440     JMP PMOV60
5CFC            8460 PMOV31
5CFC A200       8480     LDX #0      ; start the jump
5CFE 8EE05A     8500     STX JMPNDX
5D01 203C66     8520     JSR VWJUMP  ; do jump sound
5D04 4C865D     8540     JMP PMOV40
                8560 ;
                8580 ; Update X during jump
                8600 ;
5D07            8620 PMOV33
5D07 ADDB5A     8640     LDA SCRFLG
5D0A F035       8660     BEQ PMOV34  ; no X change
```

Moore Business Forms, Inc. sv 1412RH

```
5D0C 20E25A   8700      JSR TRACK
5D0F A5D7     8720      LDA R+3     ; check for jumping over intrsctn
5D11 2940     8740      AND #$40
5D13 F007     8760      BEQ PM0332
5D15 A908     8780      LDA #NPATH  ; force path if intrsctn
5D17 85DC     8800      STA R+8
5D19 4C415D   8820      JMP PMOV34
5D1C          8840 PM0332
5D1C A5EF     8860      LDA R+27    ; check for change of dir. while jumping
5D1E C904     8880      CMP #4
5D20 9004     8900      BCC PM0333  ; check for illegal track
5D22 C9FD     8920      CMP #-3
5D24 9007     8940      BCC PM0334
5D26          8960 PM0333
5D26 ADD85A   8980      LDA BLSTNZR
5D29 45EF     9000      EOR R+27    ; sign change?
5D2B 1014     9020      BPL PMOV34  ; no, ok to track - branch
5D2D          9040 PM0334
5D2D ADD95A   9060      LDA BUGDIR  ; else keep same dir.
5D30 85D6     9080      STA R+2
5D32 ADA842   9100      LDA BUGX
5D35 85D4     9120      STA R+0
5D37 A901     9140      LDA #1
5D39 85DC     9160      STA R+8
5D3B 20E25A   9180      JSR TRACK
5D3E 4C465D   9200      JMP PM0342
5D41          9220 PMOV34
5D41 A5D6     9240      LDA R+2
5D43 8DD95A   9260      STA BUGDIR
              9280 ;
              9300 ; Sequence jump pictures
              9320 ;
5D46          9340 PM0342
5D46 CEDC5A   9360      DEC JMPCTR  ; delay between jump moves
5D49 F003     9380      BEQ PMOV35
5D4B 4C885D   9400      JMP PMOV40
5D4E          9420 PMOV35
5D4E A90C     9440      LDA #JMPDLY
5D50 8DDC5A   9460      STA JMPCTR
5D53 E8       9480      INX         ; inc for"next jump step
5D54 E02D     9500      CPX #ENDJMP ; end of sequence?
5D56 D02B     9520      BNE PMOV45  ; no - branch
5D58 A280     9540      LDX #128    ; else cancel jumr"index
5D5A 8EE05A   9560      STX JMPNDX
              9580 ;
              9600 ; Check for killing a truck
              9620 ;
5D5D A201     9640      LDX #OBJMAX-1
5D5F          9660 PM0400
5D5F 20D965   9680      JSR COLISN
5D62 B006     9700      BCS PM0410
5D64          9710 PM0405
5D64 CA       9720      DEX         ; check 'em all
5D65 10F8     9740      BPL PM0400
5D67 4C5D5E   9760      JMP PMOV90
5D6A          9780 PM0410
5D6A 204B64   9880      JSR TERASE  ; erase the truck
5D6D A9FF     9900      LDA #-1     ; and cancel it
```

```
5D67 4C5D5E      9760      JMP PMOV90
5D6A             9780 PMO410
5D6A 204B64      9880      JSR TERASE  ; erase the truck
5D6D A9FF        9900      LDA #-1     ; and cancel it
```



```
5D6F 9DEC62      9920      STA TYPE,X
5D72 204C66      9940      JSR TRKDIE  ; do truck die sound
                 9960 ;
                 9980 ; bonus for killing truck
                 010000 ;
5D75 A900        010020    LDA #0      ; add 400
5D77 85D9        010040    STA R+5
5D79 A904        010060    LDA #$04
5D7B 85DA        010080    STA R+6
5D7D 20B85F      010100    JSR ADDP1   ; add to score
5D80 4C645D      010120    JMP PMO405
                 010140 ;
                 010160 ; Update jump picture/position
                 010180 ;
5D83             010200 PMOV45
5D83 8EE05A      010220    STX JMPNDX
5D86             010240 PMOV40
5D86 A5D5        010260    LDA R+1     ; modify player's Y
5D88 18          010280    CLC
5D89 7D3C5C      010300    ADC JMPADD,X
5D8C A8          010320    TAY         ; save
5D8D BD695C      010340    LDA JMPPIC,X ; get picture
5D90 AED95A      010360    LDX BUGDIR  ; modify for left/right
5D93 E003        010380    CPX #3
5D95 9003        010400    BCC PMOV46
5D97 18          010420    CLC
5D98 6903        010440    ADC #3
5D9A             010460 PMOV46
5D9A 85DD        010480    STA R+9     ; set dir to display
5D9C A6D4        010500    LDX R+0
5D9E 8EA842      010520    STX BUGX    ; update player's X
5DA1 A5DC        010540    LDA R+8
5DA3 8DDA5A      010560    STA BPTHCT
5DA6 4C505E      010580    JMP PMOV80
                 010600 ;
                 010620 ; get next X position
                 010640 ;
5DA9             010660 PMOV60
5DA9 ADDB5A      010680    LDA SCRFLG  ; X only changes after scroll
5DAC D003        010700    BNE PMOV611
5DAE 4C385E      010720    JMP PMOV70  ; no turns if no scroll
5DB1             010740 PMOV611
5DB1 20E25A      010760    JSR TRACK
                 010780 ;
                 010800 ; TURNS
                 010820 ;
5DB4             010840 PMOV61
5DB4 A5D7        010860    LDA R+3     ; check road status (returned from TRACK
                           call)
5DB6 2940        010880    AND #$40    ; intersection?
5DB8 F07E        010900    BEQ PMOV70  ; branch if not a turning point
5DBA AD00D3      010920    LDA PORTA
5DBD 290C        010940    AND #$0C
5DBF C90C        010960    CMP #$0C
5DC1 D02A        010980    BNE PMO628  ; branch if turning
                 011000 ;
                 011020 ; coast across intersection
```

Moore Business Forms, Inc. sv 1412RH

```
                    011040 ;
5DC3 AD0C40         011060  LDA SCRSPD  ; special case is dir 3, fwd ...
5DC6 300F           011080  BMI PM0625  ; check fwd
5DC8 A5D7           011100  LDA R+3     ; check can turn
5DCA 2902           011120  AND #UR
5DCC F009           011140  BEQ PM0625
5DCE A5D6           011160  LDA R+2     ; check dir
5DD0 C903           011180  CMP #3
5DD2 D003           011200  BNE PM0625
5DD4 4C2E5E         011220  JMP PMOV68  ; force turn
5DD7                011240 PM0625
5DD7 AD0C40         011260  LDA SCRSPD  ; ... or dir 0, bwd
5DDA 105C           011280  BPL PMOV70
5DDC A5D7           011300  LDA R+3
5DDE 2908           011320  AND #LR
5DE0 F056           011340  BEQ PMOV70
5DE2 A5D6           011360  LDA R+2
5DE4 D052           011380  BNE PMOV70
5DE6 E6D4           011400  INC R+0     ; fudge to align X
5DE8 E6D4           011420  INC R+0
5DEA 4C2E5E         011440  JMP PMOV68  ; force turn
                    011460 ;
5DED                011480 PM0628
5DED 2908           011500  AND #8      ; turning right?
5DEF D025           011520  BNE PMOV64  ; no - branch
5DF1 AD0C40         011540  LDA SCRSPD  ; going forward or backward?
5DF4 300B           011560  BMI PMOV62
5DF6 A5D7           011580  LDA R+3     ; can make this turn?
5DF8 2902           011600  AND #UR
5DFA F03C           011620  BEQ PMOV70  ; no - branch
5DFC A903           011640  LDA #3      ; forward, right
5DFE 4C2E5E         011660  JMP PMOV68
5E01                011680 PMOV62
5E01 A5D7           011700  LDA R+3
5E03 2908           011720  AND #LR
5E05 F031           011740  BEQ PMOV70
5E07 A5D6           011760  LDA R+2     ; turning right, bwd needs fudge
                                          correction
5E09 C903           011780  CMP #3
5E0B D004           011800  BNE PMOV63
5E0D E6D4           011820  INC R+0
5E0F E6D4           011840  INC R+0
5E11                011860 PMOV63
5E11 A900           011880  LDA #0      ; backward, right
5E13 4C2E5E         011900  JMP PMOV68
5E16                011920 PMOV64
5E16 AD0C40         011940  LDA SCRSPD
5E19 300B           011960  BMI PMOV66
5E1B A5D7           011980  LDA R+3
5E1D 2901           012000  AND #UL
5E1F F017           012020  BEQ PMOV70
5E21 A900           012040  LDA #0      ; forward, left
5E23 4C2E5E         012060  JMP PMOV68
5E26                012080 PMOV66
5E26 A5D7           012100  LDA R+3
5E28 2904           012120  AND #LL
5E2A F00C           012140  BEQ PMOV70
5E2C A903           012160  LDA #3      ; backward, left
5E2E                012180 PMOV68
```

```
5E2A F00C        012120    AND #LL
5E2C A903        012140    BEQ PMOV70
5E2E             012160    LDA #3      ; backward, left
                 012180 PMOV68
```


D1:TRACK3.M65

```
5E2E 85D6        012200    STA R+2     ; change direction
5E30 A908        012220    LDA #NPATH  ; force path to get across intersection
5E32 8DDA5A      012240    STA BPTHCT
5E35 4C3D5E      012260    JMP PMOV75
5E38             012280 PMOV70
5E38 A5DC        012300    LDA R+8
5E3A 8DDA5A      012320    STA BPTHCT
5E3D             012340 PMOV75
5E3D A5D4        012360    LDA R+0     ; update X
5E3F 8DA842      012380    STA BUGX
5E42 AA          012400    TAX
5E43 A5D5        012420    LDA R+1
5E45 8DA942      012440    STA BUGY
5E48 A8          012460    TAY
5E49 A5D6        012480    LDA R+2
5E4B 8DD95A      012500    STA BUGDIR
5E4E 85DD        012520    STA R+9
5E50             012540 PMOV80
5E50 A5EE        012560    LDA R+26
5E52 8DD75A      012580    STA BLSTINC
5E55 A5EF        012600    LDA R+27
5E57 8DD85A      012620    STA BLSTNZR
                 012640    ;
5E5A 20BA5E      012660    JSR SHOCAR  ; display VW at new position
                 012680 ;
5E5D             012700 PMOV90
5E5D 60          012720    RTS
                 012740 ;
                 012760 ; TABLE OF STATUS BYTES FOR ROAD
                 012780 ; CHARACTERS
                 012800 ; bits:
                 012820 ;    Road choices
                 012840 ; 0  Upper Left
                 012860 ; 1  Upper Right
                 012880 ; 2  Lower Left
                 012900 ; 3  Lower Right
                 012920 ;
                 012940 ; 4,5  grade
                 012960 ;        = 00 flat
                 012980 ;        = 01 uphill
                 013000 ;        = 10 downhill
                 013020 ; 6  intersection
                 013040 ;        = 0  no intersct.
                 013060 ;        = 1  intersection
                 013080 ; 7  flags a char. which
                 013100 ;      vehicles cannot be on
                 013120 ;      Then if bit 0 =0 , ground
                 013140 ;                  =1 , water
                 013160 ;
5E5E             013180 RDSTAT1
5E5E 80000003    013200    .BYTE 128,0,0,3,3,$43,$4C,129
5E62 03434C81
5E66 81808080    013220    .BYTE 129,128,128,128,128,128,0,0
5E6A 80800000
5E6E 0303434C    013240    .BYTE 3,3,$43,$4C,16+1,16+1,16+1,16+1
5E72 11111111
5E76 80801111    013260    .BYTE 128,128,16+1,16+1,16+1,128,32+2,32+2
```

Moore Business Forms, Inc.

```
5E7E 22252525   013280  .BYTE 32+2,32+5,32+5,32+5,128,128,128,128
5E82 80808080
5E86 80808080   013300  .BYTE 128,128,128,128,128,128
5E8A 8080
                013320 ; Second table indicates which line
                013340 ; of char. turning pt. occurs
5E8C            013360 RDSTAT2
5E8C 00000000   013380  .BYTE 0,0,0,0,0,15,0,0
5E90 000F0000
5E94 00000000   013400  .BYTE 0,0,0,0,0,0,0,0
5E98 00000000
5E9C 00000F00   013420  .BYTE 0,0,15,0,0,0,0,0
5EA0 00000000
5EA4 00000000   013440  .BYTE 0,0,0,0,0,0,0,0
5EA8 00000000
5EAC 00000000   013460  .BYTE 0,0,0,0,0,0,0,0
5EB0 00000000
5EB4 00000000   013480  .BYTE 0,0,0,0,0,0
5EB8 0000
                013500 ;
    =0001       013520 UL =  $01
    =0002       013540 UR =  $02
    =0004       013560 LL =  $04
    =0008       013580 LR =  $08
                013600 ;
                013620 ;
                013640 ;
                013660 ;
                013680 ; DRAW THE CAR AS A P/M GRAPHIC
                013700 ; Xreg and Yreg hold car's X,Y
5EBA            013720 SHOCAR
                013740 ; Direction to display is R+9
                013760 ;
5EBA 8A         013780  TXA          ; convert to P/M  X coord.
5EBB 18         013800  CLC
5EBC 6930       013820  ADC #48
5EBE 8DE55F     013840  STA BUGH0    ; set X
5EC1 8DE65F     013860  STA BUGH1
5EC4 6908       013880  ADC #8
5EC6 8DE75F     013900  STA BUGH2
5EC9 8DE85F     013920  STA BUGH3
                013940 ;
                013960 ; Update Y
                013980 ;
5ECC 98         014000  TYA          ; convert to P/M Y coord.
5ECD 0A         014020  ASL A
5ECE 18         014040  CLC
                014060 ;
5ECF 6920       014080  ADC #32
5ED1 A8         014100  TAY
5ED2 CCDD5A     014120  CPY OLDBUGY ; change in Y?
5ED5 D00A       014140  BNE SHOC03  ; yes - erase and redraw
5ED7 A5DD       014160  LDA R+9     ; change in direction?
5ED9 CDDE5A     014180  CMP OLDBDIR
5EDC D02D       014200  BNE SHOC07  ; yes - redraw but don't erase
5EDE 4C3F5F     014220  JMP SHOC20  ; exit without rewriting P/M
                014240 ;
                014260 ; erase old BUG
```

Moore Business Forms, Inc sv 14128H

```
5EDE 4C3F5F     014220   JMP SHOC20  ; exit without rewriting P/M
                014240 ;
                014260 ; erase old BUG
```



```
                014280 ;
5EE1            014300 SHOC03
5EE1 98         014320   TYA
5EE2 ACDD5A     014340   LDY OLDBUGY
5EE5 8DDD5A     014360   STA OLDBUGY ; save new Y pos
5EE8 A900       014380   LDA #0
5EEA A205       014400   LDX #5      ; erase 5 top andbottom byte of old image
5EEC            014420 SHOC05
5EEC 990074     014440   STA P0,Y
5EEF 991B74     014460   STA P0+27,Y
5EF2 990075     014480   STA P1,Y
5EF5 991B75     014500   STA P1+27,Y
5EF8 990076     014520   STA P2,Y
5EFB 991B76     014540   STA P2+27,Y
5EFE 990077     014560   STA P3,Y
5F01 991B77     014580   STA P3+27,Y
5F04 C8         014600   INY
5F05 CA         014620   DEX
5F06 D0E4       014640   BNE SHOC05
5F08 ACDD5A     014660   LDY OLDBUGY ; restore new Y
5F0B            014680 SHOC07
                014700 ;
5F0B A5DD       014720   LDA R+9     ; form index to VW P/M data
5F0D C906       014740   CMP #6      ; check dir.
5F0F 9003       014760   BCC SHOC08
5F11 4C425F     014780   JMP SHOC30  ; don't draw invalid dir.
5F14            014800 SHOC08
5F14 8DDE5A     014820   STA OLDBDIR
5F17 AA         014840   TAX
5F18 BD435F     014860   LDA M32TBL,X ; *32
5F1B AA         014880   TAX
5F1C 18         014900   CLC
5F1D 6920       014920   ADC #32     ; #bytes per P/M
5F1F 85D4       014940   STA R+0
                014960 ;
5F21            014980 SHOC10
5F21 BDE95F     015000   LDA BUGP0,X
5F24 990074     015020   STA P0,Y
5F27 BDA960     015040   LDA BUGP1,X
5F2A 990075     015060   STA P1,Y
5F2D BD6961     015080   LDA BUGP2,X
5F30 990076     015100   STA P2,Y
5F33 BD2962     015120   LDA BUGP3,X
5F36 990077     015140   STA P3,Y
5F39 C8         015160   INY
5F3A E8         015180   INX
5F3B E4D4       015200   CPX R+0
5F3D D0E2       015220   BNE SHOC10
5F3F            015240 SHOC20
5F3F 204B5F     015260   JSR WHEELS  ; spin the wheels
5F42            015280 SHOC30
5F42 60         015300   RTS
                015320 ;
                015340 ;
5F43            015360 M32TBL ;      *32 table
5F43 00204060   015380   .BYTE 0,32,64,96,128,160,192,224
5F47 80A0C0E0
```

```
                    015400 ;
                    015420 ;
                    015440 ;
                    015460 ; DO WHEEL SPIN
                    015480 ;
5F4B                015500 WHEELS
5F4B ADDB5A         015520   LDA SCRFLG  ; only spin while scrolling
5F4E F05E           015540   BEQ WHEE20
5F50 ADE05A         015560   LDA JMPNDX  ; no spin while jumping
5F53 1059           015580   BPL WHEE20
5F55 A5DD           015600   LDA R+9     ; choose data depending on direction
5F57 297F           015620   AND #$7F
5F59 C903           015640   CMP #3
5F5B 900E           015660   BCC WHEE05
5F5D 38             015680   SEC         ; facing right
5F5E E903           015700   SBC #3
5F60 A0C1           015720   LDY #$C1
5F62 84D5           015740   STY R+1
5F64 A0A0           015760   LDY #$A0
5F66 84DD           015780   STY R+9
5F68 4C735F         015800   JMP WHEE10
5F6B                015820 WHEE05
5F6B A005           015840   LDY #$05    ; facing left
5F6D 84D5           015860   STY R+1
5F6F A083           015880   LDY #$83
5F71 84DD           015900   STY R+9
5F73                015920 WHEE10
5F73 0A             015940   ASL A
5F74 0A             015960   ASL A
5F75 85D4           015980   STA R+0
5F77 CEE15A         016000   DEC WHLSEQ
5F7A ADE15A         016020   LDA WHLSEQ  ; wheel sequence ctr
5F7D 2901           016040   AND #1      ; 2 positions for wheels
5F7F 05D4           016060   ORA R+0
5F81 AA             016080   TAX
5F82 ADA942         016100   LDA BUGY
5F85 18             016120   CLC
5F86 6910           016140   ADC #16
5F88 0A             016160   ASL A
5F89 85D7           016180   STA R+3
5F8B 7DAF5F         016200   ADC WHLPOS,X ; carry clear
5F8E A8             016220   TAY
5F8F A5D5           016240   LDA R+1
5F91 990074         016260   STA P0,Y
5F94 A5DD           016280   LDA R+9
5F96 990076         016300   STA P2,Y
5F99 8A             016320   TXA
5F9A 4901           016340   EOR #1
5F9C 05D4           016360   ORA R+0
5F9E AA             016380   TAX
5F9F BDAF5F         016400   LDA WHLPOS,X
5FA2 18             016420   CLC
5FA3 65D7           016440   ADC R+3
5FA5 A8             016460   TAY
5FA6 A900           016480   LDA #0
5FA8 990074         016500   STA P0,Y
5FAB 990076         016520   STA P2,Y
5FAE                016540 WHEE20
5FAE 60             016560   RTS
```

Moore Business Forms, Inc. style 14128H

```
5FAB 990076     016520   STA P2,Y
5FAE            016540 WHEE20
5FAE 60         016560   RTS
```



```
                016580 ;
5FAF            016600 WHLPOS ;      Y offset for wheel marks
5FAF 1A1B1C1D   016620   .BYTE 26,27,28,29 ; level
5FB3 1B1C1D1E   016640   .BYTE 27,28,29,30 ; up
5FB7 1A1B1C1D   016660   .BYTE 26,27,28,29 ; down
                016680 ;
                016700 ;
                016720 ;
                016740 ; ADD TO PLAYER1'S SCORE
                016760 ;
5FBB            016780 ADDP1
5FBB AD0240     016800   LDA PSCOR0
5FBE 85D6       016820   STA R+2
5FC0 AD0340     016840   LDA PSCOR1
5FC3 85D7       016860   STA R+3
5FC5 AD0440     016880   LDA PSCOR2
5FC8 85D8       016900   STA R+4
5FCA A9BC       016920   LDA # <HMEM+6
5FCC 85D4       016940   STA R+0
5FCE A947       016960   LDA # >HMEM+6
5FD0 85D5       016980   STA R+1
5FD2 203C53     017000   JSR ADDSCOR
5FD5 A5D6       017020   LDA R+2
5FD7 8D0240     017040   STA PSCOR0
5FDA A5D7       017060   LDA R+3
5FDC 8D0340     017080   STA PSCOR1
5FDF A5D8       017100   LDA R+4
5FE1 8D0440     017120   STA PSCOR2
5FE4 60         017140   RTS
                017160 ;
5FE5            017180 BUGH0 *= *+1
5FE6            017200 BUGH1 *= *+1
5FE7            017220 BUGH2 *= *+1
5FE8            017240 BUGH3 *= *+1
                015760   .INCLUDE #D2:PMCAR.M65
                0900     .PAGE "PM DATA"
```

Moore Business Forms, Inc. sv 1417RH

```
                        1000 ; DATA FOR PLAYER'S CAR
                        1020 ; CAR IS A PLAYER MISSILE GRAPHIC
5FE9                    1040 BUGP0
                        1060 ; P0    LEFT, LEVEL
5FE9 00010307           1080     .BYTE 0,1,3,7,7,15,59,59
5FED 070F3B3B
5FF1 0B39697D           1100     .BYTE 11,57,105,125,117,23,27,27
5FF5 75171B1B
5FF9 584F4F4F           1120     .BYTE 88,79,79,79,10,3,1,1
5FFD 0A030101
6001 04040404           1140     .BYTE 4,4,4,4,4,4,0,0
6005 04040000
                        1160 ; P0   LEFT, UP
6009 307E7E38           1180     .BYTE 48,126,126,56,57,63,63,55
600D 393F3F37
6011 1B193D2D           1200     .BYTE 27,25,61,45,44,40,10,10
6015 2C280A0A
6019 02050506           1220     .BYTE 2,5,5,6,6,15,3,3
601D 060F0303
6021 07050504           1240     .BYTE 7,5,5,4,4,4,0,0
6025 04040000
                        1260 ; P0  LEFT, DOWN
6029 00000001           1280     .BYTE 0,0,0,1,3,3,7,7
602D 03030707
6031 070D0909           1300     .BYTE 7,13,9,9,9,57,59,15
6035 09393B0F
6039 347B7F7F           1320     .BYTE 52,123,127,127,27,19,17,81
603D 1B131151
6041 44444404           1340     .BYTE 68,68,68,4,4,4,0,0
6045 04040000
                        1360 ; P0   RIGHT, LEVEL
6049 00070F0F           1380     .BYTE 0,7,15,15,31,31,31,63
604D 1F1F1F3F
6051 33212121           1400     .BYTE 51,33,33,33,33,33,33,127
6055 2121217F
6059 55D4FFFD           1420     .BYTE 85,212,255,253,60,253,193,193
605D 3CFDC1C1
6061 00000000           1440     .BYTE 0,0,0,0,0,0,0,0
6065 00000000
                        1460 ; P0   RIGHT, UP
6069 0C1F1F1C           1480     .BYTE 12,31,31,28,31,31,63,63
606D 1F1F3F3F
6071 3F3F3F3F           1500     .BYTE 63,63,63,63,63,63,51,33
6075 3F3F3321
6079 21617FD5           1520     .BYTE 33,97,127,213,213,254,253,253
607D D5FEFDFD
6081 FDC1C100           1540     .BYTE 253,193,193,0,0,0,0,0
6085 00000000
                        1560 ; P0   RIGHT, DOWN
6089 00000007           1580     .BYTE 0,0,0,7,15,15,31,31
608D 0F0F1F1F
6091 33212121           1600     .BYTE 51,33,33,33,33,33,33,127
6095 2121217F
6099 5554FFFF           1620     .BYTE 85,84,255,255,253,253,193,193
609D FDFDC1C1
60A1 00000000           1640     .BYTE 0,0,0,0,0,0,0,0
60A5 00000000
60A9                    1660 BUGP1
```

Moore Business Forms, Inc. ev 14129H

```
60A1 00000000   1640      .BYTE 0,0,0,0,0,0,0,0
60A5 00000000
60A9            1660 BUGP1

PAGE 51   UPN DOWN - for Atari systems
PM DATA                                          D2:PMCAR.M65

                1680 ; P1   LEFT, LEVEL
60A9 00000001   1700      .BYTE 0,0,0,1,3,3,51,49
60AD 03033331
60B1 11063672   1720      .BYTE 17,6,54,114,154,248,236,236
60B5 9AF8ECEC
60B9 EFF6F6F3   1740      .BYTE 239,246,246,243,247,109,14,14
60BD F76D0E0E
60C1 0F0F0F0F   1760      .BYTE 15,15,15,15,15,15,7,6
60C5 0F0F0706
                1780 ; P1   LEFT, UP
60C9 00062624   1800      .BYTE 0,6,38,36,22,80,81,88
60CD 16505158
60D1 6C646A7A   1820      .BYTE 108,100,106,122,123,127,125,61
60D5 7B7F7D3D
60D9 0D0E0E0B   1840      .BYTE 13,14,14,11,11,10,13,13
60DD 0B0A0D0D
60E1 0D0E0E0F   1860      .BYTE 13,14,14,15,15,15,7,6
60E5 0F0F0706
                1880 ; P1   LEFT, DOWN
60E9 00000000   1900      .BYTE 0,0,0,0,0,1,3,3
60ED 00010303
60F1 01000002   1920      .BYTE 1,0,0,2,6,54,52,16
60F5 06363410
60F9 0B24769B   1940      .BYTE 11,36,118,155,245,236,238,238
60FD F5ECEEEE
6101 FFFFFFFF   1960      .BYTE 255,255,255,255,111,15,7,6
6105 6F0F0706
                1980 ; P1   RIGHT, LEVEL
6109 0000070F   2000      .BYTE 0,0,7,15,15,31,31,19
610D 0F1F1F13
6111 0101000E   2020      .BYTE 1,1,0,14,30,30,30,0
6115 1E1E1E00
6119 2A2B0082   2040      .BYTE 42,43,0,130,195,2,62,62
611D C3023E3E
6121 FFFFF1F1   2060      .BYTE 255,255,241,241,241,240,96,96
6125 F1F06060
                2080 ; P1   RIGHT UP
6129 00030302   2100      .BYTE 0,3,3,2,0,7,31,31
612D 00071F1F
6131 3F3F3F1E   2120      .BYTE 63,63,63,30,12,0,12,30
6135 0C000C1E
6139 1E1E002A   2140      .BYTE 30,30,0,42,42,1,195,195
613D 2A01C3C3
6141 C33E30F1   2160      .BYTE 195,62,48,241,241,241,225,96
6145 F1F1E160
                2180 ; P1   RIGHT, DOWN
6149 00000000   2200      .BYTE 0,0,0,0,7,15,15,19
614D 070F0F13
6151 01000E1E   2220      .BYTE 1,0,14,30,30,30,30,0
6155 1E1E1E00
6159 2A2B00C1   2240      .BYTE 42,43,0,193,195,2,62,62
615D C3023E3E
6161 FFFFFDFD   2260      .BYTE 255,255,253,253,241,240,224,96
6165 F1F0E060
6169            2280 BUGP2
                2300 ; P2   LEFT, LEVEL
6169 00E0F0F0   2320      .BYTE 0,224,240,240,248,248,248,252
```

Moore Business Forms, Inc. av 1412RH

```
6171 CC848484   2340  .BYTE 204,132,132,132,132,132,132,254
6175 848484FE
6179 AA2BFFBF   2360  .BYTE 170,43,255,191,60,191,131,131
617D 3CBF8383
6181 00000000   2380  .BYTE 0,0,0,0,0,0,0,0
6185 00000000
                2400 ; P2  LEFT, UP
6189 30F8F838   2420  .BYTE 48,248,248,56,248,248,252,252
618D F8F8FCFC
6191 FCFCFCFC   2440  .BYTE 252,252,252,252,252,252,204,132
6195 FCFCCC84
6199 8486FEAB   2460  .BYTE 132,134,254,171,171,127,191,191
619D AB7FBFBF
61A1 BF838300   2480  .BYTE 191,131,131,0,0,0,0,0
61A5 00000000
                2500 ; P2  LEFT, DOWN
61A9 000000E0   2520  .BYTE 0,0,0,224,240,240,248,248
61AD F0F0F8F8
61B1 CC848484   2540  .BYTE 204,132,132,132,132,132,132,254
61B5 848484FE
61B9 AA2AFFFF   2560  .BYTE 170,42,255,255,191,191,131,131
61BD BFBF8383
61C1 00000000   2580  .BYTE 0,0,0,0,0,0,0,0
61C5 00000000
                2600 ; P2  RIGHT, LEVEL
61C9 0080C0E0   2620  .BYTE 0,128,192,224,224,240,220,220
61CD E0F0DCDC
61D1 D09C96BE   2640  .BYTE 208,156,150,190,174,232,216,216
61D5 AEE8D8D8
61D9 1AF2F2F2   2660  .BYTE 26,242,242,242,80,192,128,128
61DD 50C08080
61E1 20202020   2680  .BYTE 32,32,32,32,32,32,0,0
61E5 20200000
                2700 ; P2  RIGHT, UP
61E9 0C7E7E1C   2720  .BYTE 12,126,126,28,156,252,252,236
61ED 9CFCFCEC
61F1 D898BCB4   2740  .BYTE 216,152,188,180,52,20,80,80
61F5 34145050
61F9 40A0A060   2760  .BYTE 64,160,160,96,96,240,192,192
61FD 60F0C0C0
6201 E0A0A020   2780  .BYTE 224,160,160,32,32,32,0,0
6205 20200000
                2800 ; P2  RIGHT, DOWN
6209 000000B0   2820  .BYTE 0,0,0,128,192,192,224,224
620D C0C0E0E0
6211 E0B09090   2840  .BYTE 224,176,144,144,144,156,220,240
6215 909CDCF0
6219 2CDEFEFE   2860  .BYTE 44,222,254,254,216,200,136,138
621D D8C8888A
6221 22222220   2880  .BYTE 34,34,34,32,32,32,0,0
6225 20200000
6229            2900 BUGP3
                2920 ; P3  LEFT, LEVEL
6229 0000E0F0   2940  .BYTE 0,0,224,240,240,248,248,200
622D F0F8F8C8
6231 80800070   2960  .BYTE 128,128,0,112,120,120,120,0
6235 78787800
6239 54D40041   2980  .BYTE 84,212,0,65,195,64,124,124
```

Moore Business Forms, Inc. sv 1412RH

```
622D F0F8F8C8
6231 80800070    2960      .BYTE 128,128,0,112,120,120,120,0
6235 78787800
6239 54D40041    2980      .BYTE 84,212,0,65,195,64,124,124
```



```
623D C3407C7C
6241 FFFF8F8F    3000      .BYTE 255,255,143,143,143,15,6,6
6245 8F0F0606
                 3020 ; P3   LEFT, UP
6249 00C0C040    3040      .BYTE 0,192,192,64,0,224,248,248
624D 00E0F8F8
6251 FCFCFC78    3060      .BYTE 252,252,252,120,48,0,48,120
6255 30003078
6259 78780054    3080      .BYTE 120,120,0,84,84,128,195,195
625D 5480C3C3
6261 C37C0C8F    3100      .BYTE 195,124,12,143,143,143,135,6
6265 8F8F8706
                 3120 ; P3   LEFT, DOWN
6269 00000000    3140      .BYTE 0,0,0,0,224,240,240,200
626D E0F0F0C8
6271 80007078    3160      .BYTE 128,0,112,120,120,120,120,0
6275 78787800
6279 54D40083    3180      .BYTE 84,212,0,131,195,64,124,124
627D C3407C7C
6281 FFFFBFBF    3200      .BYTE 255,255,191,191,143,15,7,6
6285 8F0F0706
                 3220 ; P3   RIGHT, LEVEL
6289 00000080    3240      .BYTE 0,0,0,128,192,192,204,140
628D C0C0CC8C
6291 88606C4E    3260      .BYTE 136,96,108,78,89,31,55,55
6295 591F3737
6299 F76F6FCF    3280      .BYTE 247,111,111,207,239,182,112,112
629D EFB67070
62A1 F0F0F0F0    3300      .BYTE 240,240,240,240,240,240,224,96
62A5 F0F0E060
                 3320 ; P3   RIGHT, UP
62A9 00606424    3340      .BYTE 0,96,100,36,104,10,138,26
62AD 680A8A1A
62B1 3626565E    3360      .BYTE 54,38,86,94,222,254,190,188
62B5 DEFEBEBC
62B9 B07070D0    3380      .BYTE 176,112,112,208,208,80,176,176
62BD D050B0B0
62C1 B07070F0    3400      .BYTE 176,112,112,240,240,240,224,96
62C5 F0F0E060
                 3420 ; P3   RIGHT, DOWN
62C9 00000000    3440      .BYTE 0,0,0,0,0,128,192,192
62CD 0080C0C0
62D1 80000040    3460      .BYTE 128,0,0,64,96,108,44,8
62D5 606C2C08
62D9 D0246ED9    3480      .BYTE 208,36,110,217,175,55,119,119
62DD AF377777
62E1 FFFFFFFF    3500      .BYTE 255,255,255,255,246,240,224,96
62E5 F6F0E060
                 3600 ;    END OF PLAYER P/M DATA
                 015780    .INCLUDE #D2:MOVE.M65
                 1000      .PAGE "MOVE"
```

Moore Business Forms, Inc. #v 1412RH

```
                    1020 ; UP'N DOWN
                    1040 ; MOVE.M65
                    1060 ;
                    1080 ; TABLES FOR OBJECT ENTRIES
                    1100 ;
      =0002         1120 OBJMAX = 2       ; max # of objects on screen
      =000E         1140 YCRSH = 14       ; Y tolerance on collision detect
      =000F         1160 XCRSH = 15
      =0014         1180 OVRLPX = 20      ; overlap tolerance on X coord.
      =0014         1200 OVRLPY = 20      ;    "          "    Y
 62E9               1220 XPRIME *= *+1
 62EA               1240 REDRAW *= *+1    ; redraw overlapped object flag
 62EB               1260 ONTOP *= *+1     ; obj# of object on top of overlapped
 62EC               1280 TYPE *= *+9
 62F5               1300 XPOS *= *+9
 62FE               1320 YPOS *= *+9
 6307               1340 DIR *=  *+9
 6310               1360 SPEED *= *+9
 6319               1380 SPDCTR *= *+9
 6322               1400 MODE *= *+9
 632B               1420 PATH *= *+9
 6334               1440 LASTINC *= *+9
 633D               1460 LASTNZR *= *+9
 6346               1480 OVRLAY *= *+9
 634F               1500 DRAWN *= *+9
                    1520 ;
                    1540 ; ROUTINE TO MOVE ALL OBJECTS IN
                    1560 ; THE OBJECT TABLE
                    1580 ;
 6358               1600 MOVE
 6358 A201          1620      LDX #OBJMAX-1 ; max # objects
 635A               1640 MOVE10
 635A BDEC62        1660      LDA TYPE,X    ; active entry?
 635D 1004          1680      BPL MOVE16    ; yes - branch
 635F               1700 MOVE15
 635F CA            1720      DEX
 6360 10F8          1740      BPL MOVE10
 6362 60            1760      RTS
 6363               1780 MOVE16
 6363 BD1063        1800      LDA SPEED,X ; time to move?
 6366 1005          1820      BPL MOVE20
 6368 49FF          1840      EOR #$FF
 636A 18            1860      CLC
 636B 6901          1880      ADC #1
 636D               1900 MOVE20
 636D 18            1920      CLC
 636E 7D1963        1940      ADC SPDCTR,X
 6371 9D1963        1960      STA SPDCTR,X
 6374 90E9          1980      BCC MOVE15    ; no move - branch
                    2000 ;
 6376 A9FF          2020      LDA #$FF      ; default flag no redraw
 6378 8DEA62        2040      STA REDRAW
 637B A001          2060      LDY #OBJMAX-1 ; check for overlaps
 637D 207164        2080      JSR OVRLAP
 6380 C0FF          2100      CPY #$FF
 6382 F03F          2120      BEQ MOVE25    ; none - branch
 6384 B9EE62        2140      LDA YPOS,Y    ; check for interfere with track
 6387 38            2160      SEC
```

Moore Business Forms, Inc. sv 1412RH

```
6380 C0FF      2100      CPY #$FF
6382 F03F      2120      BEQ MOVE25  ; none - branch
6384 B9FE62    2140      LDA YPOS,Y  ; check for interfere with track
6387 38        2160      SEC
```



```
6388 FDFE62    2180      SBC YPOS,X
638B 1005      2200      BPL MOVE22
638D 49FF      2220      EOR #$FF
638F 18        2240      CLC
6390 6901      2260      ADC #1
6392           2280 MOVE22
6392 C909      2300      CMP #OVRLPY-YTRKPT
6394 B01A      2320      BCS MOVE23  ; doesn't interfere
6396 ECEB62    2340      CPX ONTOP
6399 D01A      2360      BNE MOVE24
639B 8CEA62    2380      STY REDRAW
639E 20095A    2400      JSR ERASE
63A1 8EEB62    2420      STX ONTOP
63A4 AEEA62    2440      LDX REDRAW
63A7 20095A    2460      JSR ERASE
63AA AEEB62    2480      LDX ONTOP
63AD 4CC663    2500      JMP MOVE26
63B0           2520 MOVE23
63B0 ECEB62    2540      CPX ONTOP   ; if this obj on top no need to erase
                                      other
63B3 F00E      2560      BEQ MOVE25
63B5           2580 MOVE24
63B5 8EEB62    2600      STX ONTOP
63B8 98        2620      TYA
63B9 AA        2640      TAX
63BA 20095A    2660      JSR ERASE   ; erase other
63BD 8EEA62    2680      STX REDRAW
63C0 AEEB62    2700      LDX ONTOP
63C3           2720 MOVE25
63C3 20095A    2740      JSR ERASE   ; erase this truck
63C6           2760 MOVE26
63C6 BCFE62    2780      LDY YPOS,X  ; move it
63C9 BD1063    2800      LDA SPEED,X
63CC 1004      2820      BPL MOVE30
63CE C8        2840      INY
63CF 4CD363    2860      JMP MOVE40
63D2           2880 MOVE30
63D2 88        2900      DEY
63D3           2920 MOVE40
63D3 C060      2940      CPY #96     ; get rid of trucks that move off screen
63D5 B05E      2960      BCS MOVE50
63D7 84D5      2980      STY R+1
63D9 BDF562    3000      LDA XPOS,X
63DC C9A0      3020      CMP #160
63DE B055      3040      BCS MOVE50
63E0 85D4      3060      STA R+0
63E2 BD0763    3080      LDA DIR,X
63E5 85D6      3100      STA R+2
63E7 BD2B63    3120      LDA PATH,X
63EA 85DC      3140      STA R+8
63EC BD3463    3160      LDA LASTINC,X
63EF 85EE      3180      STA R+26
63F1 BD3D63    3200      LDA LASTNZR,X
63F4 85EF      3220      STA R+27
63F6 BD1063    3240      LDA SPEED,X
63F9 85F0      3260      STA R+28
               3280 ;
```

Moore Business Forms, Inc. sv 1472H

```
63FB 20E25A     3300     JSR TRACK
```



```
                3320 ;
                3340 ; update new parms
                3360 ;
63FE A5D4       3380     LDA R+0
6400 9DE562     3400     STA XPOS,X
6403 A5D5       3420     LDA R+1
6405 9DFE62     3440     STA YPOS,X
6408 A5D6       3460     LDA R+2
640A 9D0763     3480     STA DIR,X
640D A5DC       3500     LDA R+8
640F 9D2B63     3520     STA PATH,X
6412 A5EE       3540     LDA R+26
6414 9D3463     3560     STA LASTINC,X
6417 A5EF       3580     LDA R+27
6419 9D3D63     3600     STA LASTNZR,X
                3620 ;
641C ADEA62     3640     LDA REDRAW
641F 3009       3660     BMI MOVE45
6421 AEEA62     3680     LDX REDRAW
6424 20C059     3700     JSR DRAW
6427 AEEB62     3720     LDX ONTOP
642A            3740 MOVE45
642A 20C059     3760     JSR DRAW    ; draw at new position
642D 8EEB62     3780     STX ONTOP
6430 B003       3800     BCS MOVE50  ; draw not successful - branch
6432 4C5F63     3820     JMP MOVE15
6435            3840 MOVE50
6435 A9FF       3860     LDA #-1     ; cancel object
6437 9DEC62     3880     STA TYPE,X
643A 204B64     3900     JSR TERASE
643D 4C5F63     3920     JMP MOVE15
                3940 ;
                3960 ; INIT THE OBJECT TABLES
                3980 ;
6440            4000 IOBJTB
6440 A9FF       4020     LDA #-1     ; inactive flag value
6442 A201       4040     LDX #OBJMAX-1 ; # entries
6444            4060 IOBJ10
6444 9DEC62     4080     STA TYPE,X
6447 CA         4100     DEX
6448 10FA       4120     BPL IOBJ10
                4140 ;
644A 60         4160     RTS
                4180 ;
                4200 ; Erase truck according to
                4220 ; ONTOP priority
                4240 ;
644B            4260 TERASE
644B ECEB62     4380     CPX ONTOP
644E F01D       4400     BEQ TERA10
                4420 ;
6450 8EE962     4421     STX XPRIME
6453 8A         4422     TXA
6454 4901       4424     EOR #1
6456 8DEB62     4426     STA ONTOP
6459 AA         4440     TAX
645A 20095A     4460     JSR ERASE
645D AEE962     4480     LDX XPRIME
```

Moore Business Forms, Inc. sv 1412RH

```
6459 AA          4440      TAX
645A 20095A      4460      JSR ERASE
645D AEE962      4480      LDX XPRIME
```



```
6460 20095A      4500      JSR ERASE
6463 AEEB62      4520      LDX ONTOP
6466 20C059      4540      JSR DRAW
6469 AEE962      4560      LDX XPRIME
646C 60          4580      RTS
646D             4600 TERA10
646D 20095A      4620      JSR ERASE
6470 60          4640      RTS
                 4660 ;
                 4680 ;
                 4700 ; See if trucks overlap
                 4720 ; ENTER:
                 4740 ;   Yreg = start index
                 4760 ; EXIT:
                 4780 ;   Yreg = # of first overlap
                 4800 ;        = $FF if no overlap
                 4820 ; Uses R+18
                 4840 ;
6471             4860 OVRLAP
6471 86E6        4880      STX R+18
                 4900 ;
6473             4920 OVRL05
6473 B9EC62      4940      LDA TYPE,Y ; check for entries
6476 3029        4960      BMI OVRL40
                 4980 ;;   LDA DRAWN,Y ; can't overlap if not drawn
                 5000 ;;   BEQ OVRL40
6478 C4E6        5020      CPY R+18   ; can't ovrlap self
647A F025        5040      BEQ OVRL40
                 5060 ;
647C B9F562      5080      LDA XPOS,Y
647F 38          5100      SEC
6480 FDF562      5120      SBC XPOS,X
6483 1005        5140      BPL OVRL10
6485 49FF        5160      EOR #$FF
6487 18          5180      CLC
6488 6901        5200      ADC #1
648A             5220 OVRL10
648A C914        5240      CMP #OVRLPX
648C B013        5260      BCS OVRL40  ; no ovrlap
                 5280 ;
648E B9FE62      5300      LDA YPOS,Y
6491 38          5320      SEC
6492 FDFE62      5340      SBC YPOS,X
6495 1005        5360      BPL OVRL20
6497 49FF        5380      EOR #$FF
6499 18          5400      CLC
649A 6901        5420      ADC #1
649C             5440 OVRL20
649C C914        5460      CMP #OVRLPY
649E B001        5480      BCS OVRL40
64A0 60          5500      RTS         ;found an overlap
64A1             5520 OVRL40
64A1 88          5540      DEY
64A2 10CF        5560      BPL OVRL05
64A4 60          5580      RTS
                 5600 ;
                 5620 ;
```

Moore Business Forms, Inc. sv 1412RH

```
                        5640 ; INIT AN ENEMY TRUCK
                        5660 ; Enter:
                        5680 ;      TPNDX = # of type to init (0-5)
                        5700 ;      TPROB = (ROUND-1)*6
                        5720 ;      R+0,1,7,8,9,10
                        5740 ; USES:
                        5760 ;
 64A5                   5780 TINIT
                        5800 ; Check probablities to see if to init truck
                        5820 ;
 64A5 ACD865            5840       LDY TPNDX    ; get index to truck parameters
 64A8 A206              5860       LDX #NTYPES
 64AA                   5880 TINI10
 64AA B9AE65            5900       LDA TPRQB,Y ; get probability of init
 64AD F005              5920       BEQ TINI15
 64AF CD0AD2            5940       CMP RANDOM  ; init this type?
 64B2 B005              5960       BCS TINI20  ; yes - branch
 64B4                   5980 TINI15
 64B4 C8                6000       INY
 64B5 CA                6020       DEX
 64B6 D0F2              6040       BNE TINI10
 64B8 60                6060       RTS         ; no inits this time
                        6080 ;
                        6100 ; Find an open slot
                        6120 ;
 64B9                   6140 TINI20
 64B9 A201              6160       LDX #OBJMAX-1
                        6180 ;;    LDX #1      ;;;;DEBUG
 64BB                   6200 TINI30
 64BB BDEC62            6220       LDA TYPE,X
 64BE 3004              6240       BMI TINI40
 64C0 CA                6260       DEX
 64C1 10F8              6280       BPL TINI30
 64C3 60                6300       RTS         ; no open slots
                        6320 ;
                        6340 ; Only init if VW overtakes/overtaken
                        6360 ;
 64C4                   6380 TINI40
 64C4 98                6400       TYA
 64C5 48                6420       PHA
 64C6 8A                6440       TXA
 64C7 48                6460       PHA
                        6480 ;
                        6500 ; Check speeds to see who overtakes
                        6520 ;
 64C8 AD0AD2            6540       LDA RANDOM  ; init spd from tables
 64CB 39CC65            6560       AND TMAXSP,Y
 64CE 18                6580       CLC
 64CF 79D265            6600       ADC TMINSP,Y
 64D2 85E1              6620       STA R+13
                        6640 ;
 64D4 A5E1              6660       LDA R+13    ; check signs
 64D6 4D0C40            6680       EOR SCRSPD
 64D9 301B              6700       BMI TIN553  ; opposite dir - branch
 64DB A5E1              6720       LDA R+13    ; check magnitudes
 64DD 1005              6740       BPL TINI50
 64DF 49FF              6760       EOR #$FF
 64E1 18                6780       CLC
 64E2 6901              6800       ADC #1
```

Moore Business Forms, Inc. sv 1412RH

```
64DB A5E1       6720     LDA R+13    ; check magnitudes
64DD 1005       6740     BPL TINI50
64DF 49FF       6760     EOR #$FF
64E1 18         6780     CLC
64E2 6901       6800     ADC #1
```



```
64E4            6820 TINI50
64E4 85DC       6840     STA R+8
64E6 AD0C40     6860     LDA SCRSPD
64E9 1005       6880     BPL TINI52
64EB 49FF       6900     EOR #$FF
64ED 18         6920     CLC
64EE 6901       6940     ADC #1
64F0            6960 TINI52
64F0 C5DC       6980     CMP R+8
64F2 900C       7000     BCC TIN552  ; VW is slower - truck will overtake
64F4 F014       7020     BEQ TINI54  ; same speed
64F6            7040 TIN553
64F6 AD0C40     7060     LDA SCRSPD  ; VW is faster
64F9 300A       7080     BMI TIN550
64FB            7100 TIN551
64FB A903       7120     LDA #3      ; init at top
64FD 4C1265     7140     JMP TINI56
6500            7160 TIN552
6500 AD0C40     7180     LDA SCRSPD
6503 30F6       7200     BMI TIN551
6505            7220 TIN550
6505 A95F       7240     LDA #95     ;  "  bottom
6507 4C1265     7260     JMP TINI56
650A            7280 TINI54
650A AD0AD2     7300     LDA RANDOM  ; select top/bot randomly
650D 10F6       7320     BPL TIN550
650F 4CFB64     7340     JMP TIN551
6512            7360 TINI56
6512 85DB       7380     STA R+7     ; save
6514 18         7400     CLC
6515 690B       7420     ADC #YTRKPT ; find road at track point
6517 6D0A40     7440     ADC DSCROL
651A AA         7460     TAX
651B BD3E40     7480     LDA DSPTBL,X ; set ptr to line
651E 85D4       7500     STA R+0
6520 BD1D41     7520     LDA DSPTBH,X
6523 85D5       7540     STA R+1
                7560 ; Put truck on road
                7580 ; Search for road
                7600 ;
6525 A027       7620     LDY #39     ; 40 bytes/line
6527 A200       7640     LDX #0
6529            7660 TINI60
6529 B1D4       7680     LDA (R+0),Y
652B C9FF       7700     CMP #$FF    ; look for road on this line
652D D029       7720     BNE TINI70
652F 88         7740     DEY         ; must get at least 2 road bytes to be
                           sure
6530 D1D4       7760     CMP (R+0),Y
6532 D024       7780     BNE TINI70
6534 88         7800     DEY         ; now find next non-road byte
6535 D1D4       7820     CMP (R+0),Y
6537 D005       7840     BNE TINI62
6539 88         7860     DEY         ; 4 road bytes is too many - can't be road
653A D1D4       7880     CMP (R+0),Y
653C F01A       7900     BEQ TINI70
653E            7920 TINI62
```

Moore Business Forms, Inc. sv 1412RH

```
653E A903       7940     LDA #3       ; find left pixel of road
```



```
6540 85EE       7960     STA R+26
6542 B1D4       7980     LDA (R+0),Y
6544            8000 TINI64
6544 6A         8020     ROR A
6545 9008       8040     BCC TINI66
6547 6A         8060     ROR A
6548 9005       8080     BCC TINI66
654A C6EE       8100     DEC R+26
654C 4C4465     8120     JMP TINI64
654F            8140 TINI66
654F 98         8160     TYA          ; convert to Xpos
6550 0A         8180     ASL A
6551 0A         8200     ASL A
6552 18         8220     CLC
6553 65EE       8240     ADC R+26
6555 95D6       8260     STA R+2,X    ; save
6557 E8         8280     INX
6558            8300 TINI70
6558 88         8320     DEY
6559 10CE       8340     BPL TINI60
655B E000       8360     CPX #0       ; find any roads?
655D F00B       8380     BEQ TINI74   ; no - branch
655F 86DA       8400     STX R+6      ; save # roads found
                8420 ;
6561 AD0AD2     8440     LDA RANDOM   ; figure out which Xpos to use
6564 2903       8460     AND #$03     ; max #roads found=4
                8480 ;
6566 C5DA       8500     CMP R+6      ; valid road?
6568 9005       8520     BCC TINI80   ; yes - branch
656A            8540 TINI74
656A 68         8560     PLA          ; can't init
656B AA         8580     TAX
656C 68         8600     PLA
656D A8         8620     TAY
656E 60         8640     RTS
                8660 ;
656F            8680 TINI80
656F AA         8700     TAX
6570 B5D6       8720     LDA R+2,X
6572 85DA       8740     STA R+6      ; save initial Xpos
                8760 ;
6574 68         8780     PLA
6575 AA         8800     TAX
6576 68         8820     PLA
6577 A8         8840     TAY
                8860 ;
                8880 ;
                8900 ; Init truck's parameters
                8920 ;
6578 A5DA       8940     LDA R+6
657A 9DF562     8960     STA XPOS,X
657D A5DB       8980     LDA R+7
657F 9DFE62     9000     STA YPOS,X
6582 A5E1       9020     LDA R+13
6584 9D1063     9040     STA SPEED,X
6587 98         9060     TYA
6588 9DEC62     9080     STA TYPE,X
658B A901       9100     LDA #1
```

Moore Business Forms, Inc. 14128H

```
6587 98          9060      TYA
6588 9DEC62      9080      STA TYPE,X
658B A901        9100      LDA #1
```



```
658D 9D0763      9120      STA DIR,X
6590 9D3463      9140      STA LASTINC,X
6593 9D3D63      9160      STA LASTNZR,X
6596 A900        9180      LDA #0
6598 9D2B63      9200      STA PATH,X
                 9220 ;
659B 20E25A      9240      JSR TRACK
659E A001        9260      LDY #OBJMAX-1
65A0 207164      9280      JSR OVRLAP
65A3 C0FF        9300      CPY #$FF
65A5 F003        9320      BEQ TINI90
65A7 8EEB62      9340      STX ONTOP
65AA             9360 TINI90
65AA 20C059      9380      JSR DRAW    ; initial draw
65AD 60          9400      RTS
                 9420 ;
                 9440 ; TABLE OF PROBABILITIES TO INIT TRUCKS
                 9460 ; PROB is x/255 per call to TINIT
65AE             9480 TPROB
65AE 01000000    9500      .BYTE 1,0,0,0,0,0 ; ROUND 1
65B2 0000
65B4 01000000    9520      .BYTE 1,0,0,0,0,0 ; ROUND 2
65B8 0000
65BA 01000000    9540      .BYTE 1,0,0,0,0,0 ;     3
65BE 0000
65C0 00000505    9560      .BYTE 0,0,5,5,5,0 ;     4
65C4 0500
65C6 00000005    9580      .BYTE 0,0,0,5,5,5 ;     5
65CA 0505
                 9600 ; Initial speeds for trucks
                 9620 ;
65CC             9640 TMAXSP ;          truck max speed (ANDed)
65CC 07070707    9660      .BYTE 7,7,7,7,7,7
65D0 0707
65D2             9680 TMINSP ;          truck min speed (ADDed)
65D2 01010101    9700      .BYTE 1,1,1,1,1,1
65D6 0101
                 9720 ;
65D8             9740 TPNDX *= *+1
    =0006        9760 NTYPES = 6        ; number of truck types
                 9780 ;
                 9800 ; COLLISION ROUTINE
                 9820 ; CHECKS FOR COLLISIONS BETWEEN
                 9840 ; PLAYER'S VW AND ENEMY TRUCKS
                 9860 ;
                 9880 ; Returns:
                 9900 ;      carry set and Xreg has index
                 9920 ;      of truck collided with ,
                 9940 ; or
                 9960 ;      carry clear if no collision
                 9980 ;
65D9             010000 COLISN
                 010020 ;;LDX #OBJMAX-1 ; check all OBJTBL entries
65D9             010040 COLI10
65D9 BDEC62      010060   LDA TYPE,X   ; find active entry
65DC 3024        010080   BMI COLI40
65DE BDFE62      010100   LDA YPOS,X   ; check Y
```

Moore Business Forms, Inc. #V 1412N

```
65E1 38          010120   SEC
```



```
65E2 EDA942      010140   SBC BUGY
65E5 B005        010160   BCS COLI20  ; find absolute difference
65E7 49FF        010180   EOR #$FF
65E9 18          010200   CLC
65EA 6901        010220   ADC #1
65EC             010240 COLI20
65EC C90E        010260   CMP #YCRSH  ; Y tolerance on collision
65EE B012        010280   BCS COLI40  ; no hit - branch
                 010300 ;
65F0 BDF562      010320   LDA XPOS,X  ; check X
65F3 38          010340   SEC
65F4 EDA842      010360   SBC BUGX
65F7 B005        010380   BCS COLI30
65F9 49FF        010400   EOR #$FF
65FB 18          010420   CLC
65FC 6901        010440   ADC #1
65FE             010460 COLI30
65FE C90F        010480   CMP #XCRSH  ; X tolerance
6600 9005        010500   BCC COLI50  ; found a collision - branch
6602             010520 COLI40
6602 CA          010540   DEX         ; next object
6603 10D4        010560   BPL COLI10
6605 18          010580   CLC         ; no collisions
6606 60          010600   RTS
6607             010620 COLI50
6607 38          010640   SEC         ; return with collided truck index in
                                      Xreg.
6608 60          010660   RTS
                 010680 ;
                 010700 ;
                 010720 ; INIT PARAMS FOR NEXT ROUND
                 010740 ;
6609             010760 NEWRND
6609 F8          010770   SED
660A AD0540      010780   LDA ROUND
660D 18          010790   CLC
660E 6901        010792   ADC #1
6610 8D0540      010800   STA ROUND   ; inc round ctr
6613 D8          010810   CLD
6614 8D0548      010820   STA HMEM+79 ; display it
6617 0A          010840   ASL A       ; TPNDX=ROUND*6
6618 85D4        010860   STA R+0
661A 0A          010880   ASL A
661B 18          010900   CLC
661C 65D4        010920   ADC R+0
661E 8DD865      010940   STA TPNDX
                 010960 ;
6621 A90A        010962   LDA #10     ; init flag counter
6623 8D036F      010964   STA FLGCNT
                 010966 ;
6626 A900        010980   LDA #0      ; flag playfield VBI
6628 8D5C50      011000   STA BPAGEF  ; (not bonus page)
662B 8D5450      011020   STA TIMER   ; reset timer
662E 8D5550      011040   STA TIMER+1
6631 A93C        011042   LDA #60
6633 8D0140      011044   STA TICS
6636 A901        011060   LDA #1
6638 8D0040      011080   STA TIMRON  ; start timer
```

Moore Business Forms, Inc. 1412RH

```
6633 8D0140     011044    STA TICS
6636 A901       011060    LDA #1
6638 8D0040     011080    STA TIMRON  ; start timer
```



```
                011082 ;
663B 60         011100    RTS
                015800    .INCLUDE #D2:SOUND.M65
                1000      .PAGE "SOUND"
```

Moore Business Forms, Inc. sv 1412RH

```
                        1020 ; SOUND.M65
                        1040 ;
                        1060 ; UPN DOWN SOUND ROUTINES AND DATA
                        1080 ;
                        1100 ;
                        1120 ; THE FOLLOWING ROUTINES INIT THE
                        1140 ; VARIOUS SOUNDS
                        1160 ;
   663C                 1180 VWJUMP
663C A928               1200     LDA #$28
663E 8D0169             1220     STA AUDV3
6641 A9D2               1240     LDA # <SSTR1
6643 8DE968             1260     STA VOICE3
6646 A967               1280     LDA # >SSTR1
6648 8DEA68             1300     STA VOICE3+1
664B 60                 1320     RTS
                        1340 ;
   664C                 1360 TRKDIE
664C A928               1380     LDA #$28
664E 8D0369             1400     STA AUDV4
6651 A9AE               1420     LDA # <SSTR2
6653 8DEB68             1440     STA VOICE4
6656 A967               1460     LDA # >SSTR2
6658 8DEC68             1480     STA VOICE4+1
665B 60                 1500     RTS
                        1520 ;
   665C                 1540 VWDIE
665C A9A8               1560     LDA #$A8
665E 8DFD68             1580     STA AUDV1
6661 A9F6               1600     LDA # <SSTR3
6663 8DE568             1620     STA VOICE1
6666 A967               1640     LDA # >SSTR3
6668 8DE668             1660     STA VOICE1+1
666B A9A8               1680     LDA #$A8
666D 8DFF68             1700     STA AUDV2
6670 A928               1720     LDA # <SSTR32
6672 8DE768             1740     STA VOICE2
6675 A968               1760     LDA # >SSTR32
6677 8DE868             1780     STA VOICE2+1
667A 60                 1800     RTS
                        1820 ;
   667B                 1840 FLAG
667B A9A8               1860     LDA #$A8
667D 8D0369             1880     STA AUDV4
6680 A95A               1900     LDA # <SSTR4
6682 8DEB68             1920     STA VOICE4
6685 A968               1940     LDA # >SSTR4
6687 8DEC68             1960     STA VOICE4+1
668A 60                 1980     RTS
                        2000 ;
                        2020 ;
   668B                 2040 BSOUND
668B A928               2060     LDA #$28
668D 8D08D2             2080     STA AUDCTL
6690 A9A8               2100     LDA #$A8
6692 8DFD68             2120     STA AUDV1
6695 8D0369             2140     STA AUDV4
6698 A9A0               2160     LDA #$A0
```

Moore Business Forms, Inc. sv 14128H

```
6692 8DFD68       2120      STA AUDV1
6695 8D0369       2140      STA AUDV4
6698 A9A0         2160      LDA #$A0
```



```
669A 8D0169       2180      STA AUDV3
669D A926         2200      LDA # <SSTR5
669F 8DE568       2220      STA VOICE1
66A2 A962         2240      LDA # <SSTR5A
66A4 8DEB68       2260      STA VOICE4
66A7 A988         2280      LDA # <SSTR5B
66A9 8DE968       2300      STA VOICE3
66AC A967         2320      LDA # >SSTR5
66AE 8DE668       2340      STA VOICE1+1
66B1 A967         2360      LDA # >SSTR5A
66B3 8DEC68       2380      STA VOICE4+1
66B6 A967         2400      LDA # >SSTR5B
66B8 8DEA68       2420      STA VOICE3+1
66BB 60           2440      RTS
                  2460 ;
66BC              2480 ALLFLGS
66BC A9A8         2500      LDA #$A8
66BE 8DFD68       2520      STA AUDV1
66C1 A9CC         2540      LDA # <SSTR6
66C3 8DE568       2560      STA VOICE1
66C6 A966         2580      LDA # >SSTR6
66C8 8DE668       2600      STA VOICE1+1
66CB 60           2620      RTS
                  2640 ;
                  2660 ; GOT ALL FLAGS SOUND
66CC              2680 SSTR6
66CC 28022302     2700      .BYTE 40,2,35,2,31,2,29,2,0,16,35,8
66D0 1F021D02
66D4 00102308
66D8 2D060002     2720      .BYTE 45,6,0,2,45,6,0,2,40,6,0,2,35,6,0,2
66DC 2D060002
66E0 28060002
66E4 23060002
66E8 21060002     2740      .BYTE 33,6,0,2,35,6,0,2,40,6,0,2,33,4,0,2
66EC 23060002
66F0 28060002
66F4 21040002
66F8 23080002     2760      .BYTE 35,8,0,2,40,4,0,2,33,6,0,2,35,6,0,2
66FC 28040002
6700 21060002
6704 23060002
6708 2F100008     2780      .BYTE 47,16,0,8,53,6,0,2,60,6,0,2,60,6,0,2
670C 35060002
6710 3C060002
6714 3C060002
6718 35040002     2800      .BYTE 53,4,0,2,47,8,0,2,40,8,45,8
671C 2F080002
6720 28082D08
6724 0000         2820      .BYTE 0,0
                  2840 ;
                  2860 ; BONUS PAGE SOUND
6726              2880 SSTR5
6726 79100020     2900      .BYTE 121,16,0,32
672A 60100020     2920      .BYTE 96,16,0,32
672E 44080004     2940      .BYTE 68,8,0,4
6732 00044806     2960      .BYTE 0,4,72,6,0,2,81,8,0,8
6736 00025108
```

Moore Business Forms, Inc. sv 1412RH

```
673C 60180020   2980      .BYTE 96,24,0,32
6740 44080008   3000      .BYTE 68,8,0,8,72,6,0,2,81,8,0,8,96,6,0,2
6744 48060002
6748 51080008
674C 60060002
6750 44080008   3020      .BYTE 68,8,0,8,72,6,0,2,81,8,0,8,121,8,0,28
6754 48060002
6758 51080008
675C 7908001C
6760 0000       3040      .BYTE 0,0
6762            3060 SSTR5A
6762 23100020   3080      .BYTE 35,16,0,32,26,16,0,32
6766 1A100020
676A 150C000C   3100      .BYTE 21,12,0,12,26,12,0,12
676E 1A0C000C
6772 23100020   3120      .BYTE 35,16,0,32,26,16,0,32
6776 1A100020
677A 2A0C000C   3140      .BYTE 42,12,0,12,35,12,0,12
677E 230C000C
6782 35100020   3160      .BYTE 53,16,0,32
6786 0000       3180      .BYTE 0,0
6788            3200 SSTR5B
6788 49100020   3220      .BYTE 73,16,0,32,186,16,0,32
678C BA100020
6790 360C000C   3240      .BYTE 54,12,0,12,186,12,0,12
6794 BA0C000C
6798 AD100020   3260      .BYTE 173,16,0,32,186,16,0,32
679C BA100020
67A0 6D0C000C   3280      .BYTE 109,12,0,12,173,12,0,12
67A4 AD0C000C
67A8 74100020   3300      .BYTE 116,16,0,32
67AC 0000       3320      .BYTE 0,0
                3340 ; TRUCK DIE SOUND
67AE            3360 SSTR2
67AE 09010801   3380      .BYTE $09,1,$08,1,$07,1,$06,1
67B2 07010601
67B6 05010401   3400      .BYTE $05,1,$04,1,$03,1,$02,1
67BA 03010201
67BE 0001       3420      .BYTE 0,1
67C0 09010801   3440      .BYTE $09,1,$08,1,$07,1,$06,1
67C4 07010601
67C8 05010401   3460      .BYTE $05,1,$04,1,$03,1,$02,1
67CC 03010201
67D0 0000       3480      .BYTE 0,0
                3500 ;
                3520 ; VW JUMP SOUND
67D2            3540 SSTR1
67D2 12011001   3560      .BYTE $12,1,$10,1,$0F,1,$0E,1
67D6 0F010E01
67DA 0D010C01   3580      .BYTE $0D,1,$0C,1,$0B,1,$0A,1
67DE 0B010A01
67E2 09010801   3600      .BYTE $09,1,$08,1,$07,1,$06,1
67E6 07010601
67EA 05010401   3620      .BYTE $05,1,$04,1,$03,1,$02,1
67EE 03010201
67F2 01000000   3640      .BYTE 1,0,0,0
                3660 ;
                3680 ; VW DIE SOUND
```

```
67EE 03010201
67F2 01000000   3640      .BYTE 1,0,0,0
                3660 ;
                3680 ; VW DIE SOUND
```



```
67F6            3700 SSTR3
67F6 3C040004   3720      .BYTE 60,4,0,4,60,4,0,4,47,4,0,4
67FA 3C040004
67FE 2F040004
6802 28040004   3740      .BYTE 40,4,0,4,33,4,0,4,35,4,0,4
6806 21040004
680A 23040004
680E 28040004   3760      .BYTE 40,4,0,4,47,4,0,4,60,4,0,4
6812 2F040004
6816 3C040004
681A 3C040004   3780      .BYTE 60,4,0,4,68,4,0,4,60,4,0,4
681E 44040004
6822 3C040004
6826 0000       3800      .BYTE 0,0
6828            3820 SSTR32
6828 60050003   3840      .BYTE 96,5,0,3,96,5,0,3,81,5,0,3
682C 60050003
6830 51050003
6834 48050003   3860      .BYTE 72,5,0,3,69,5,0,3,72,5,0,3
6838 45050003
683C 48050003
6840 51050003   3880      .BYTE 81,5,0,3,96,5,0,3,123,5,0,3
6844 60050003
6848 7B050003
684C 7B050003   3900      .BYTE 123,5,0,3,123,5,0,3,123,5,0,3
6850 7B050003
6854 7B050003
6858 0000       3920      .BYTE 0,0
                3940 ;
                3960 ; FLAG SOUND
685A            3980 SSTR4
685A 30022C02   4000      .BYTE $30,2,$2C,2,$28,2,$24,2,$20,2,$1C,2
685E 28022402
6862 20021C02
6866 18021402   4020      .BYTE $18,2,$14,2,$10,2,$08,2
686A 10020802
686E 0000       4040      .BYTE 0,0
                4060 ;
                4080 ;
                4100 ; ROUTINE TO PLAY SOUND STRINGS
                4120 ;
                4140 ; THERE ARE 4 VOICES AVAILABLE
                4160 ; UNLESS TWO ARE CONNECTED VIA
                4180 ; AUDCTL.
                4200 ;
                4220 ; A SOUND STRING HAS THE FORM:
                4240 ;     freq, interval, freq, interval, ... 0,0
                4260 ; INTERVALS ARE 1/60th SECONDS.
                4280 ; A FREQ OF 0 WILL CAUSE A REST
                4300 ; FOR THAT INTERVAL.
                4320 ; STRINGS MUST BE TERMINATED BY 0,0.
                4340 ;
                4360 ; TO START A STRING:
                4380 ;     LOAD AUDV1-4 WITH AUDC1-4 VALUE.
                4400 ;     SET VOICE1-4 TO POINT TO THE STRING.
                4420 ;     SINCE A NON-ZERO IN THE HIGH
                4440 ;     BYTE WILL TRIGGER PLAYING LOAD THE
```

Moore Business Forms, Inc. sv 1412RM

```
                        4460 ;     LOW BYTE FIRST (NOTE: STRINGS
                        4480 ;     CANNOT BE ON ZERO PAGE).
                        4500 ;
                        4520 ; EXAMPLE:
                        4540 ;     LDA #$A8       ; for pure tones
                        4560 ;     STA AUDV1      ; for voice 1
                        4580 ;     LDA #<SSTRING ; low byte
                        4600 ;     STA VOICE1
                        4620 ;     LDA #>SSTRING ; high byte
                        4640 ;     STA VOICE1+1   ; init playing
                        4660 ;
                        4680 ; WHEN A STRING IS FINISHED, ITS
                        4700 ; VPTRx+1 WILL BE SET TO 0.
                        4720 ;
 6870                   4740 SFX
 6870 A200              4760     LDX #0
 6872 BDE668            4780 LP405 LDA VOICE1+1,X ; check for new sound string pointer
 6875 F010              4800     BEQ BR463   ; if zero then voice is not active
 6877 9DEE68            4820     STA VPTR1+1,X ; else get new pointer
 687A BDE568            4840     LDA VOICE1,X
 687D 9DED68            4860     STA VPTR1,X
 6880 A900              4880     LDA #0
 6882 9DE668            4900     STA VOICE1+1,X
 6885 F014              4920     BEQ BR420   ; begin playing
 6887 BDEE68            4940 BR463 LDA VPTR1+1,X ; get high byte of voice pointer
 688A F008              4960     BEQ BR419   ; 0 means voice not active
 688C BDF568            4980     LDA VTIMR1,X ; check voice timer
 688F F00A              5000     BEQ BR420   ; if zero then voice is available
 6891 DEF568            5020 BR461 DEC VTIMR1,X ; else decrement the timer
 6894 E8                5040 BR419 INX       ; increment index for next voice
 6895 E8                5060     INX
 6896 E008              5080     CPX #8      ; do for four voices
 6898 D0D8              5100     BNE LP405
 689A 60                5120     RTS
                        5140 ;
                        5160 ;  Set pointer to sound string
                        5180 ;
 689B BDED68            5200 BR420 LDA VPTR1,X
 689E 85D4              5220     STA R+0
 68A0 BDEE68            5240     LDA VPTR1+1,X
 68A3 85D5              5260     STA R+1
                        5280 ;
                        5300 ;  Play next note
                        5320 ;
 68A5 A000              5340     LDY #0
 68A7 B1D4              5360     LDA (R+0),Y ; get new frequency
 68A9 D003              5380     BNE BR457   ; 0 means a rest
 68AB 98                5400     TYA
 68AC F00B              5420     BEQ BR458   ; unconditional jump
 68AE 9D00D2            5440 BR457 STA AUDF1,X ; play new freq.
 68B1 BDFE68            5460     LDA AUDV1+1,X ; AUDV1+1 is flag for a previous rest
 68B4 D009              5480     BNE BR459   ; ...0 means currently resting
 68B6 BDFD68            5500     LDA AUDV1,X
 68B9 9D01D2            5520 BR458 STA AUDC1,X ; restore AUDC1
 68BC 9DFE68            5540     STA AUDV1+1,X ; unflag
 68BF C8                5560 BR459 INY
 68C0 B1D4              5580     LDA (R+0),Y ; get new timer interval
 68C2 D00B              5600     BNE BR421   ; 0 means end of string
```

```
68C4 9D01D2     5620     STA AUDC1,X ; stop AUDCx
```



```
68C7 9DEE68     5640     STA VPTR1+1,X ; free voice
68CA 9DFE68     5660     STA AUDV1+1,X ; indicate rest
68CD F0C5       5680     BEQ BR419 ; unconditional jump for next voice
68CF 9DF568     5700 BR421 STA VTIMR1,X ; set timer
                5720 ;
                5740 ;  Increment the 2-byte voice pointer
                5760 ;
68D2 FEED68     5780     INC VPTR1,X
68D5 D003       5800     BNE BR423
68D7 FEEE68     5820     INC VPTR1+1,X
68DA FEED68     5840 BR423 INC VPTR1,X
68DD D003       5860     BNE BR424
68DF FEEE68     5880     INC VPTR1+1,X
68E2 18         5900 BR424 CLC
68E3 90AC       5920     BCC BR461
                5940 ;
                5960 ;
68E5            5980 VOICE1 *= *+8
68ED            6000 VPTR1 *= *+8
68F5            6020 VTIMR1 *= *+8
68FD            6040 AUDV1 *= *+8
    =D201       6060 AUDC1 = $D201
    =D200       6080 AUDF1 = $D200
    =68E7       6100 VOICE2 = VOICE1+2
    =68E9       6120 VOICE3 = VOICE2+2
    =68EB       6140 VOICE4 = VOICE3+2
    =68FF       6160 AUDV2 = AUDV1+2
    =6901       6180 AUDV3 = AUDV2+2
    =6903       6200 AUDV4 = AUDV3+2
    =68EF       6220 VPTR2 = VPTR1+2
    =68F1       6240 VPTR3 = VPTR2+2
    =68F3       6260 VPTR4 = VPTR3+2
                6280 ;
                6300 ;
    =D208       6320 AUDCTL = $D208
    =D20F       6340 SKCTL = $D20F
    =0232       6360 SSKCTL = $0232
                6380 ;
                6400 ; INIT SOUND
                6420 ;
6905            6440 ISFX
                6460 ;
6905 A207       6480     LDX #7
6907 A900       6500     LDA #0
6909            6520 ISFX10
6909 9DE568     6540     STA VOICE1,X
690C 9DF568     6560     STA VTIMR1,X
690F 9DED68     6580     STA VPTR1,X
6912 9DFD68     6600     STA AUDV1,X
6915 CA         6620     DEX
6916 10F1       6640     BPL ISFX10
                6660 ;
6918 A900       6680     LDA #0
691A 8D08D2     6700     STA AUDCTL
691D A903       6720     LDA #3
691F 8D0FD2     6740     STA SKCTL
6922 8D3202     6760     STA SSKCTL
```

Moore Business Forms, Inc. sv 1412RH

```
6925 60         6780     RTS
```



```
                015820   .INCLUDE #D2:SCROLL.M65
                1000     .PAGE "SCROLL"
```

Moore Business Forms, Inc.-sv 4412RH

```
                    1020 ;
                    1040 ; SCROLL.M65
                    1060 ; SCROLL ROUTINE
                    1080 ;
6926                1100 SCROLL
6926 A900           1120     LDA #0
6928 8DDB5A         1140     STA SCRFLG  ; flag no scroll
692B AD0C40         1160     LDA SCRSPD  ; get scroll speed
692E 1005           1180     BPL SCR008  ; make positive
6930 49FF           1200     EOR #$FF
6932 18             1220     CLC
6933 6901           1240     ADC #1
6935                1260 SCR008
6935 18             1280     CLC
6936 6D0B40         1300     ADC SCRCTR
6939 8D0B40         1320     STA SCRCTR
693C 9050           1340     BCC SCR030  ; if no carry , no scroll this time
693E AD0C40         1360     LDA SCRSPD  ; flag scroll
6941 8DDB5A         1380     STA SCRFLG
                    1400 ;
                    1420 ; adjust Y of objects on screen
                    1440 ;
6944 A201           1460     LDX #OBJMAX-1
6946                1480 SCR022
6946 BDEC62         1500     LDA TYPE,X  ; check object
6949 3020           1520     BMI SCR244  ; branch if not active
694B BCFE62         1540     LDY YPOS,X
694E AD0C40         1560     LDA SCRSPD  ; add or subtract?
6951 1004           1580     BPL SCR023
6953 88             1600     DEY
6954 4C5869         1620     JMP SCR024
6957                1640 SCR023
6957 C8             1660     INY
6958                1680 SCR024
6958 C060           1700     CPY #96
695A 900B           1720     BCC SCR242
695C 20095A         1740     JSR ERASE   ; off screen
695F A9FF           1760     LDA #-1
6961 9DEC62         1780     STA TYPE,X  ; cancel
6964 4C6B69         1800     JMP SCR244
6967                1820 SCR242
6967 98             1840     TYA
6968 9DFE62         1860     STA YPOS,X
696B                1880 SCR244
696B CA             1900     DEX
696C 10D8           1920     BPL SCR022
                    1940 ;
696E                1960 SCR233
696E AD0BD4         1980     LDA VCOUNT  ; sync to screen
6971 C964           2000     CMP #100
6973 90F9           2020     BCC SCR233
6975 AD0C40         2040     LDA SCRSPD
6978 3006           2060     BMI SCR025
697A 208F69         2080     JSR UPSCRN  ; scroll up
697D 4C8369         2100     JMP SCR027
6980                2120 SCR025
6980 20D769         2140     JSR DNSCRN  ; scroll down
```

```
                        2160 ;
                        2180 ;
                        2200 ; add to score
                        2220 ;
    6983                2240 SCR027
    6983 A902           2260     LDA #2
    6985 85D9           2280     STA R+5
    6987 A900           2300     LDA #0
    6989 85DA           2320     STA R+6
    698B 20BB5F         2340     JSR ADDP1
                        2360 ;
    698E                2380 SCR030
    698E 60             2400     RTS
                        2420 ;
                        2440 ;
                        2460 ;
                        2480 ; Scrolling is accomplished by writing the new
                        2500 ; line to be scrolled in over the
                        2520 ; line about to be scrolled out and changing
                        2540 ; the display list.
                        2560 ;
                        2580 ;
                        2600 ; SCROLL THE DISPLAY WINDOW UP
                        2620 ;
    698F                2640 UPSCRN
    698F                2652 UPSCR1
    698F CE0D40         2660     DEC SCROLF  ; dec pointer to 1st playfield line, fine
    6992 1024           2680     BPL UPSC10  ; check underflow
    6994 A90F           2700     LDA #15     ; reset fine
    6996 8D0D40         2720     STA SCROLF
    6999 A5CA           2740     LDA SCROLC  ; dec coarse
    699B 38             2760     SEC
    699C E90A           2780     SBC #10     ; 10 chars/row
    699E 85CA           2800     STA SCROLC
    69A0 A5CB           2820     LDA SCROLC+1
    69A2 E900           2840     SBC #0
    69A4 85CB           2860     STA SCROLC+1
    69A6 C943           2880     CMP # >APFDATA-10U ; check underflow
    69A8 D00E           2900     BNE UPSC10
    69AA A5CA           2920     LDA SCROLC
    69AC C9AC           2940     CMP # <APFDATA-10U
    69AE D008           2960     BNE UPSC10
    69B0 A918           2980     LDA # <APFDATA+PFLEN-10U ; reset SCROLC
    69B2 85CA           3000     STA SCROLC
    69B4 A946           3020     LDA # >APFDATA+PFLEN-10U
    69B6 85CB           3040     STA SCROLC+1
                        3060 ;
                        3080 ; Update DSCROL (new top line)
                        3100 ;
    69B8                3120 UPSC10
    69B8 CE0A40         3140     DEC DSCROL  ; dec index to SMEM first line
    69BB 1005           3160     BPL UPSC15  ; branch if no underflow
    69BD A96F           3180     LDA #111    ; else reset to bottom
    69BF 8D0A40         3200     STA DSCROL
                        3220 ;
                        3240 ; Set pointer to new line destination
                        3260 ; which is the new top line.
                        3280 ;
    69C2                3300 UPSC15
```

```
                    3220 ;
                    3240 ; Set pointer to new line destination
                    3260 ; which is the new top line.
                    3280 ;
69C2                3300 UPSC15
```



```
69C2 AD0A40         3320     LDA DSCROL  ; DSCROL indexes top line of display
69C5 AA             3340     TAX         ; use to index DSPTB list of ptrs to SMEM
69C6 BD3E40         3360     LDA DSPTBL,X ; get pointer low
69C9 85D4           3380     STA R+0
69CB BD1D41         3400     LDA DSPTBH,X ;    "    high
69CE 85D5           3420     STA R+1
                    3440 ;
                    3460 ; Write PF line indexed by SCROL pointers
                    3480 ; to SMEM address in (R+0,R+1).
                    3500 ;
69D0                3520 UPSC18
69D0 205B6A         3540     JSR PFLINE
                    3560 ;
69D3 207B4F         3580     JSR CHGDSP  ; rewrite display list pointers
69D6 60             3600     RTS
                    3620 ;
                    3640 ;
                    3660 ; SCROLL THE DISPLAY WINDOW DOWN
                    3680 ;
                    3700 ; write new line to current line
                    3720 ;
69D7                3740 DNSCRN
69D7 AE0A40         3760     LDX DSCROL  ; get top line index
69DA BD3E40         3780     LDA DSPTBL,X ; set pointer
69DD 85D4           3800     STA R+0
69DF BD1D41         3820     LDA DSPTBH,X
69E2 85D5           3840     STA R+1
                    3860 ;
                    3880 ;
                    3900 ; set SCROL ptrs to 112th line
                    3920 ; after current line
                    3940 ;
69E4 A5CA           3960     LDA SCROLC  ; save SCROL ptrs
69E6 48             3980     PHA
69E7 A5CB           4000     LDA SCROLC+1
69E9 48             4020     PHA
                    4040 ;
69EA A5CA           4060     LDA SCROLC  ; add 70 bytes = 7 char rows = 112 lines
69EC 18             4080     CLC
69ED 6946           4100     ADC #70
69EF 85CA           4120     STA SCROLC
69F1 A5CB           4140     LDA SCROLC+1
69F3 6900           4160     ADC #0
69F5 85CB           4180     STA SCROLC+1
                    4200 ;
69F7 C946           4220     CMP # >APFDATA+PFLEN0 ; check ovfl
69F9 9015           4240     BCC DNSC02  ; no ovfl - branch
69FB D006           4260     BNE DNSC01  ; ovfl - branch
69FD A5CA           4280     LDA SCROLC  ; need to check low byte
69FF C922           4300     CMP # <APFDATA+PFLEN0
6A01 900D           4320     BCC DNSC02  ; no ovfl
                    4340 ;
6A03                4360 DNSC01
6A03 A5CA           4380     LDA SCROLC
6A05 38             4400     SEC
6A06 E96C           4420     SBC # <PFLEN
6A08 85CA           4440     STA SCROLC
```

Moore Business Forms, Inc. sv 1412RH

```
6A0A A5CB    4460    LDA SCROLC+1
```



```
6A0C E902    4480    SBC # >PFLEN
6A0E 85CB    4500    STA SCROLC+1
6A10         4520 DNSC02
6A10 205B6A  4540    JSR PFLINE  ; write the line
             4560 ;
6A13 68      4580    PLA         ; restore SCROL
6A14 85CB    4600    STA SCROLC+1
6A16 68      4620    PLA
6A17 85CA    4640    STA SCROLC
             4660 ;
             4680 ; Set SCROL ptrs to next line
             4700 ;
6A19 AE0D40  4720    LDX SCROLF  ; fine scroll
6A1C E8      4740    INX
6A1D 8E0D40  4760    STX SCROLF
6A20 E010    4780    CPX #16
6A22 D024    4800    BNE DNSC10
6A24 A900    4820    LDA #0
6A26 8D0D40  4840    STA SCROLF  ; reset
6A29 A5CA    4860    LDA SCROLC  ; inc coarse
6A2B 18      4880    CLC
6A2C 690A    4900    ADC #10
6A2E 85CA    4920    STA SCROLC
6A30 A5CB    4940    LDA SCROLC+1
6A32 6900    4960    ADC #0
6A34 85CB    4980    STA SCROLC+1
6A36 C946    5000    CMP # >APFDATA+PFLENU ; check overflow
6A38 D00E    5020    BNE DNSC10
6A3A A5CA    5040    LDA SCROLC
6A3C C922    5060    CMP # <APFDATA+PFLENU
6A3E D008    5080    BNE DNSC10
6A40 A9B6    5100    LDA # <PFDATA ; reset
6A42 85CA    5120    STA SCROLC
6A44 A943    5140    LDA # >PFDATA
6A46 85CB    5160    STA SCROLC+1
6A48         5180 DNSC10
             5200 ;
             5220 ; Update DSCROL (new top line)
             5240 ; the old top line now becomes
             5260 ; the new bottom line.
             5280 ;
6A48 EE0A40  5300    INC DSCROL  ; inc index to SMEM first line
6A4B AD0A40  5320    LDA DSCROL  ; check for overflow
6A4E C970    5340    CMP #112
6A50 D005    5360    BNE DNSC15
6A52 A900    5380    LDA #0      ; reset to top
6A54 8D0A40  5400    STA DSCROL
6A57         5420 DNSC15
             5440 ;
6A57 207B4F  5460    JSR CHGDSP  ; oove the display list pointers
6A5A 60      5480    RTS
             5500 ;
             5520 ;
             5540 ;
             5560 ;
             5580 ; Write the playfield line indicated
             5600 ; by SCROLC,SCROLF to the SMEM
             5620 ; location (R+0,R+1).
```

```
                    5600 ; by SCROLC,SCROLF to the SMEM
                    5620 ; location (R+0,R+1).
                    5640 ;
6A5B                5660 PFLINE
6A5B 8A             5680        TXA         ; save regs
6A5C 48             5700        PHA
6A5D 98             5720        TYA
6A5E 48             5740        PHA
6A5F A5D4           5760        LDA R+0
6A61 48             5780        PHA
6A62 A5D5           5800        LDA R+1
6A64 48             5820        PHA
6A65 A000           5840        LDY #0      ; index to chars in PF line
6A67                5860 PFLI10
6A67 98             5880        TYA
6A68 48             5900        PHA         ; save char index
6A69 B1CA           5920        LDA (SCROLC),Y
6A6B AA             5940        TAX
6A6C BD0C42         5960        LDA CHARH,X
6A6F 85D7           5980        STA R+3
6A71 BD4C42         6000        LDA CHARL,X
6A74 AE0D40         6020        LDX SCROLF
6A77 18             6040        CLC
6A78 7DFC41         6060        ADC M4TBL,X
6A7B 85D6           6080        STA R+2
6A7D A003           6100        LDY #3      ; write 4 bytes/char row
6A7F                6120 PFLI20
6A7F B1D6           6140        LDA (R+2),Y ; get char data
6A81 91D4           6160        STA (R+0),Y ; write to screen
6A83 88             6180        DEY
6A84 10F9           6200        BPL PFLI20
6A86 A5D4           6220        LDA R+0     ; update screen pointer
6A88 18             6240        CLC
6A89 6904           6260        ADC #4
6A8B 85D4           6280        STA R+0
6A8D A5D5           6300        LDA R+1
6A8F 6900           6320        ADC #0
6A91 85D5           6340        STA R+1
6A93 68             6360        PLA         ; restore char counter
6A94 A8             6380        TAY
6A95 C8             6400        INY
6A96 C00A           6420        CPY #10
6A98 D0CD           6440        BNE PFLI10  ; do next char
6A9A 68             6460        PLA
6A9B 85D5           6480        STA R+1
6A9D 68             6500        PLA
6A9E 85D4           6520        STA R+0
6AA0 68             6540        PLA
6AA1 A8             6560        TAY
6AA2 68             6580        PLA
6AA3 AA             6600        TAX
6AA4 60             6620        RTS
                    015840      .INCLUDE #D2:DATEXP.M65
                    1000        .PAGE "DATEXP"
```

Moore Business Forms, Inc. sv 1412RH

```
                    1020 ; DATEXP.M65
                    1040 ; DATA EXPANSION ROUTINE
                    1060 ;
                    1080 ;
                    1100 ; ROUTINE EXPANDS TRUCK DATA
                    1120 ; INTO RAM.  TRUCK TYPES TO EXPAND
                    1140 ; ARE PASSED IN TYPREF,TYPREF+1,...
                    1160 ; (# TYPES ALLOWED IS "NACTYPE")
                    1180 ;
                    1200 ; OBJECT DATA AND MASKS ARE THEN
                    1220 ; ACCESSED THROUGH OBJDAT AND OBJMSK
                    1240 ; WHICH ARE STRUCTURED AS FOLLOWS:
                    1260 ;
                    1280 ; TYPE=n1 DIR=0   shift0, shift1, shift2, shift3
                    1300 ;         DIR=1   shift0, shift1, shift2, shift3
                    1320 ;         DIR=3   shift0, shift1, shift2, shift3
                    1340 ;         DIR=4   shift0, shift1, shift2, shift3
                    1360 ; TYPE=n2 DIR=0   shift0, .....
                    1380 ;         DIR=1   .....
                    1382 ;
                    1390 ; Note: Dir 2,5 not used for trucks
                    1400 ;
6AA5                1420 DATEXP
6AA5 A900           1440     LDA #0      ; zero OBJREF
6AA7 A201           1460     LDX #OBJMAX-1
6AA9                1480 DATE05
6AA9 9D5A6B         1500     STA OBJREF,X
6AAC CA             1520     DEX
6AAD 10FA           1540     BPL DATE05
                    1560 ;
6AAF A200           1580     LDX #0      ; init index to TYPREF
6AB1 86DC           1600     STX R+8
6AB3                1620 DATE10
6AB3 BC586B         1640     LDY TYPREF,X ; set ptr to picture pair
6AB6 B95D6B         1660     LDA PPL,Y
6AB9 85D4           1680     STA R+0
6ABB B9696B         1700     LDA PPH,Y
6ABE 85D5           1720     STA R+1
                    1740 ;
6AC0 A900           1760     LDA #0
6AC2 85E3           1780     STA R+15    ; set fill value for shift routine
                    1800 ;
6AC4                1820 DATE15
6AC4 BD756B         1840     LDA DXPTR1L,X ; set R+2,3 to data area for shifts
6AC7 85D6           1860     STA R+2
6AC9 BD796B         1880     LDA DXPTR1H,X
6ACC 85D7           1900     STA R+3
                    1920 ;
6ACE BD7D6B         1940     LDA DXPTR2L,X ; set R+6,7 to data area for reflection
6AD1 85DA           1960     STA R+6
6AD3 BD816B         1980     LDA DXPTR2H,X
6AD6 85DB           2000     STA R+7
                    2020 ;
6AD8 8A             2040     TXA         ; save Xreg.
6AD9 48             2060     PHA
6ADA 0A             2080     ASL A       ; set ptr to master obj in OBJDAT/OBJMSK
6ADB 0A             2100     ASL A
6ADC 0A             2120     ASL A
```

Moore Business Forms, Inc. sv 1412RH

```
6ADA 0A          2080      ASL A          ; set ptr to master obj in OBJDAT/OBJMSK
6ADB 0A          2100      ASL A
6ADC 0A          2120      ASL A
```



```
6ADD 0A          2140      ASL A
6ADE 0A          2160      ASL A
6ADF AA          2180      TAX
6AE0 A5D4        2200      LDA R+0
6AE2 9D856B      2220      STA OBJDAT,X
6AE5 A5D5        2240      LDA R+1
6AE7 9D866B      2260      STA OBJDAT+1,X
                 2280 ;
6AEA 20056C      2300      JSR SHAPEX   ; expand 1st shape
6AED A5D4        2320      LDA R+0      ; set ptr to 2nd shape
6AEF 18          2340      CLC
6AF0 6950        2360      ADC #80
6AF2 85D4        2380      STA R+0
6AF4 9002        2400      BCC DATE20
6AF6 E6D5        2420      INC R+1
6AF8             2440 DATE20
6AF8 A5D6        2441      LDA R+2
6AFA 85DA        2442      STA R+6
6AFC A5D7        2443      LDA R+3
6AFE 85DB        2444      STA R+7
                 2445 ;
6B00 A5D6        2446      LDA R+2
6B02 38          2447      SEC
6B03 E930        2448      SBC #<560
6B05 85D6        2449      STA R+2
6B07 A5D7        2450      LDA R+3
6B09 E902        2451      SBC #>560
6B0B 85D7        2452      STA R+3
                 2453 ;
6B0D 20056C      2460      JSR SHAPEX   ; expand 2nd shape
6B10 68          2480      PLA          ; recover Xreg
6B11 48          2500      PHA
6B12 0A          2520      ASL A
6B13 0A          2540      ASL A
6B14 0A          2560      ASL A
6B15 0A          2580      ASL A
6B16 0A          2600      ASL A
6B17 AA          2620      TAX
6B18 A5D4        2640      LDA R+0
6B1A 9D8D6B      2660      STA OBJDAT+8,X
6B1D A5D5        2680      LDA R+1
6B1F 9D8E6B      2700      STA OBJDAT+9,X
6B22 68          2720      PLA
6B23 AA          2740      TAX
                 2760 ;
6B24 E002        2780      CPX #2       ; already do masks?
6B26 B016        2800      BCS DATE30   ; yes - branch
6B28 BC586B      2820      LDY TYPREF,X ; else setup ptr to mask data
6B2B B9636B      2840      LDA PPL+6,Y
6B2E 85D4        2860      STA R+0
6B30 B96F6B      2880      LDA PPH+6,Y
6B33 85D5        2900      STA R+1
6B35 E8          2920      INX
6B36 E8          2940      INX
6B37 A9FF        2960      LDA #$FF     ; fill value for masks
6B39 85E3        2980      STA R+15
6B3B 4CC46A      3000      JMP DATE15   ; do masks
```

Moore Business Forms, Inc. sv 1412RH

```
6B3E                3020 DATE30
```



```
6B3E A6DC           3030     LDX R+8
6B40 BC586B         3040     LDY TYPREF,X ; set OBJREF entry
6B43 8A             3060     TXA
6B44 0A             3080     ASL A       ; *32
6B45 0A             3100     ASL A
6B46 0A             3120     ASL A
6B47 0A             3140     ASL A
6B48 0A             3160     ASL A
6B49 9D5A6B         3180     STA OBJREF,X
                    3200 ;
                    3220 ;
6B4C E6DC           3240     INC R+8
6B4E A6DC           3260     LDX R+8
6B50 E002           3280     CPX #NACTYPE ; # of active object types
6B52 F003           3300     BEQ DATE40
6B54 4CB36A         3320     JMP DATE10
6B57                3340 DATE40
6B57 60             3360     RTS
                    3380 ;
6B58                3400 TYPREF *= *+2
    =0002           3420 NACTYPE = 2     ; # active objects
6B5A                3440 OBJREF *= *+OBJMAX
    =2E80           3460 MASKX = $2E80   ; size = 2240 bytes = $8c0
    =3740           3480 DATAX = $3740   ; size= 2240 = $8c0
6B5C FF             3500 FFBYTE .BYTE $FF
                    3520 ;
                    3540 ;
6B5D                3560 PPL
6B5D C30303         3580     .BYTE  <TRUCK, <SWEEPR, <FLGCAR ; data
6B60 030303         3600     .BYTE  <FLGCAR, <FLGCAR, <FLGCAR
6B63 630303         3620     .BYTE  <TRUCKM, <SWEEPRM, <FLGCARM ; mask
6B66 030303         3640     .BYTE  <FLGCARM, <FLGCARM, <FLGCARM
6B69                3660 PPH
6B69 6D6F6F         3680     .BYTE  >TRUCK, >SWEEPR, >FLGCAR ; data
6B6C 6F6F6F         3700     .BYTE  >FLGCAR, >FLGCAR, >FLGCAR
6B6F 6E6F6F         3720     .BYTE  >TRUCKM, >SWEEPRM, >FLGCARM ; mask
6B72 6F6F6F         3740     .BYTE  >FLGCARM, >FLGCARM, >FLGCARM
6B75                3760 DXPTR1L
6B75 40A0           3780     .BYTE  <DATAX, <DATAX+1120
6B77 80E0           3800     .BYTE  <MASKX, <MASKX+1120
6B79                3820 DXPTR1H
6B79 373B           3840     .BYTE  >DATAX, >DATAX+1120
6B7B 2E32           3860     .BYTE  >MASKX, >MASKX+1120
6B7D                3880 DXPTR2L
6B7D 2080           3900     .BYTE  <DATAX+480, <DATAX+1600
6B7F 60C0           3920     .BYTE  <MASKX+480, <MASKX+1600
6B81                3940 DXPTR2H
6B81 393D           3960     .BYTE  >DATAX+480, >DATAX+1600
6B83 3034           3980     .BYTE  >MASKX+480, >MASKX+1600
                    4000 ;
                    4020 ;
6B85                4040 OBJDAT
6B85 0000           4060     .WORD 0,DATAX,DATAX+80,DATAX+160
6B87 4037
6B89 9037
6B8B E037
6B8D 0000           4080     .WORD 0,DATAX+240,DATAX+320,DATAX+400
```

3C00 − 800 = 3340 − 8C0 = 2480

Moore Business Forms, Inc. sv 1412RM

```
6B8F 3038
6B91 8038
6B93 D038
6B95 2039       4100    .WORD DATAX+480,DATAX+560,DATAX+640,DATAX+720
6B97 7039
6B99 C039
6B9B 103A
6B9D 603A       4120    .WORD DATAX+800,DATAX+880,DATAX+960,DATAX+1040
6B9F B03A
6BA1 003B
6BA3 503B
                4140 ;
6BA5 0000       4160    .WORD 0,DATAX+1120,DATAX+1200,DATAX+1280
6BA7 A03B
6BA9 F03B
6BAB 403C
6BAD 0000       4180    .WORD 0,DATAX+1360,DATAX+1440,DATAX+1520
6BAF 903C
6BB1 E03C
6BB3 303D
6BB5 803D       4200    .WORD DATAX+1600,DATAX+1680,DATAX+1760,DATAX+1840
6BB7 D03D
6BB9 203E
6BBB 703E
6BBD C03E       4220    .WORD DATAX+1920,DATAX+2000,DATAX+2080,DATAX+2160
6BBF 103F
6BC1 603F
6BC3 B03F
                4240 ;
                4260 ;
6BC5            4280 OBJMSK
6BC5 0000       4300    .WORD 0,MASKX,MASKX+80,MASKX+160
6BC7 802E
6BC9 D02E
6BCB 202F
6BCD 0000       4320    .WORD 0,MASKX+240,MASKX+320,MASKX+400
6BCF 702F
6BD1 C02F
6BD3 1030
6BD5 6030       4340    .WORD MASKX+480,MASKX+560,MASKX+640,MASKX+720
6BD7 B030
6BD9 0031
6BDB 5031
6BDD A031       4360    .WORD MASKX+800,MASKX+880,MASKX+960,MASKX+1040
6BDF F031
6BE1 4032
6BE3 9032
                4380 ;
6BE5 0000       4400    .WORD 0,MASKX+1120,MASKX+1200,MASKX+1280
6BE7 E032
6BE9 3033
6BEB 8033
6BED 0000       4420    .WORD 0,MASKX+1360,MASKX+1440,MASKX+1520
6BEF D033
6BF1 2034
6BF3 7034
6BF5 C034       4440    .WORD MASKX+1600,MASKX+1680,MASKX+1760,MASKX+1840
6BF7 1035
```

Moore Business Forms, Inc. sv 1412RH

```
6BFB B035
6BFD 0036      4460      .WORD MASKX+1920,MASKX+2000,MASKX+2080,MASKX+2160
6BFF 5036
6C01 A036
6C03 F036
               4480 ;
               4500 ;
               4520 ; EXPAND THE SHAPE POINTED TO BY
               4540 ; R+0,1.  R+2,3 POINTS TO SHIFT EXPANSION
               4560 ; AREA, R+6,7 POINTS TO REFLECTION
               4580 ; EXPANSION AREA.
               4600 ;
6C05           4620 SHAPEX
6C05 A5D4      4640      LDA R+0      ; preserve R+0,1
6C07 48        4660      PHA
6C08 A5D5      4680      LDA R+1
6C0A 48        4700      PHA
               4720 ;
6C0B A901      4740      LDA #1       ; set flag to do shifts twice
6C0D 85ED      4760      STA R+25
6C0F 20916C    4780      JSR REFLECT ; reflect the master image
               4800 ;
6C12           4820 SHAP05
6C12 A903      4840      LDA #3       ; do 3 shifts of master
6C14 85DD      4860      STA R+9
6C16           4880 SHAP10
6C16 20516C    4900      JSR SHIFT
               4920 ;
6C19 A5D6      4940      LDA R+2
6C1B 85D4      4960      STA R+0
6C1D A5D7      4980      LDA R+3
6C1F 85D5      5000      STA R+1
               5020 ;
6C21 A5D6      5040      LDA R+2
6C23 18        5060      CLC
6C24 6950      5080      ADC #80
6C26 85D6      5100      STA R+2
6C28 9002      5120      BCC SHAP30
6C2A E6D7      5140      INC R+3
               5160 ;
6C2C           5180 SHAP30
6C2C C6DD      5200      DEC R+9
6C2E D0E6      5220      BNE SHAP10
               5240 ;
6C30 C6ED      5260      DEC R+25     ; see if 2nd time through
6C32 3016      5280      BMI SHAP40   ; yes - exit
6C34 A5DB      5300      LDA R+7      ; else set ptrs for shifting reflected
                                      image
6C36 85D5      5320      STA R+1
6C38 A5DA      5340      LDA R+6
6C3A 85D4      5360      STA R+0
6C3C 18        5380      CLC
6C3D 6950      5400      ADC #80
6C3F 85D6      5420      STA R+2
6C41 A5D5      5440      LDA R+1
6C43 6900      5460      ADC #0
6C45 85D7      5480      STA R+3
6C47 4C126C    5500      JMP SHAP05
```

Moore Business Forms, Inc. sv 1412RH

```
6C41 A5D5       5440      LDA R+1
6C43 6900       5460      ADC #0
6C45 85D7       5480      STA R+3
6C47 4C126C     5500      JMP SHAP05
```



```
                5520 ;
6C4A            5540 SHAP40
6C4A 68         5560      PLA
6C4B 85D5       5580      STA R+1
6C4D 68         5600      PLA
6C4E 85D4       5620      STA R+0
6C50 60         5640      RTS
                5660 ;
                5680 ;
                5700 ; SHIFT DATA 1 PIXEL TO THE RIGHT
                5720 ; SOURCE IS R+0,1 DESTINATION IS R+2,3
                5740 ; Fill byte is in R+15 (value shifted
                5760 ; in from left).
                5780 ;
6C51            5800 SHIFT
6C51 A903       5820      LDA #3
6C53 85D9       5840      STA R+5
6C55            5860 SHIF10
6C55 A000       5880      LDY #0
6C57            5900 SHIF15
6C57 A200       5920      LDX #0        ; copy row to registers
6C59            5940 SHIF20
6C59 B1D4       5960      LDA (R+0),Y
6C5B 95DE       5980      STA R+10,X
6C5D C8         6000      INY
6C5E E8         6020      INX
6C5F E005       6040      CPX #5
6C61 D0F6       6060      BNE SHIF20
6C63 A5E3       6080      LDA R+15
                6100 ;
6C65 6A         6120      ROR A
6C66 66DE       6140      ROR R+10      ; shift row right by 1 pixel
6C68 66DF       6160      ROR R+11
6C6A 66E0       6180      ROR R+12
6C6C 66E1       6200      ROR R+13
6C6E 66E2       6220      ROR R+14
6C70 6A         6240      ROR A
6C71 66DE       6260      ROR R+10
6C73 66DF       6280      ROR R+11
6C75 66E0       6300      ROR R+12
6C77 66E1       6320      ROR R+13
6C79 66E2       6340      ROR R+14
                6360 ;
6C7B 98         6380      TYA           ; rewrite registers to dest. row
6C7C 38         6400      SEC
6C7D E905       6420      SBC #5
6C7F A8         6440      TAY
                6460 ;
6C80 A200       6480      LDX #0
6C82            6500 SHIF30
6C82 B5DE       6520      LDA R+10,X
6C84 91D6       6540      STA (R+2),Y
6C86 C8         6560      INY
6C87 E8         6580      INX
6C88 E005       6600      CPX #5
6C8A D0F6       6620      BNE SHIF30
                6640 ;
```

Moore Business Forms, Inc. sv 1412RM

```
6C8C C050       6660     CPY #80      ; do for 80 bytes (16 rows)
```



```
6C8E D0C7       6680     BNE SHIF15
                6700 ;
6C90 60         6720     RTS
                6740 ;
                6760 ;
                6780 ; REFLECT DATA AT R+0,1 TO R+6,7
                6782 ; LAST COLUMN OF BYTES IS NOT
                6784 ; REFLECTED
                6800 ;
6C91            6820 REFLECT
6C91 A04F       6840     LDY #79      ; reflect 80 bytes
6C93            6860 REFL10
6C93 A900       6880     LDA #0
6C95 85D8       6900     STA R+4
                6920 ;
6C97 B1D4       6940     LDA (R+0),Y ; get byte to reflect
                6960 ;
                6980 ; Pixel bits must be switched if
                7000 ; different since ROL will switch
                7020 ; them otherwise
                7040 ;
6C99 A204       7060     LDX #4       ; reflect 4 pixels/byte
6C9B            7080 REFL20
6C9B 85EC       7100     STA R+24
6C9D 24EC       7120     BIT R+24    ; set N,V flags
6C9F 1005       7140     BPL REFL30  ; are pixel bits the same?
6CA1 7007       7160     BVS REFL50
6CA3 4CA86C     7180     JMP REFL40
6CA6            7200 REFL30
6CA6 5002       7220     BVC REFL50
6CA8            7240 REFL40
6CA8 49C0       7260     EOR ##$C0    ; need to reverse pixel
6CAA            7280 REFL50
6CAA 0A         7300     ASL A
6CAB 66D8       7320     ROR R+4
6CAD 0A         7340     ASL A
6CAE 66D8       7360     ROR R+4
6CB0 CA         7380     DEX
6CB1 D0E8       7400     BNE REFL20
                7420 ;
6CB3 98         7440     TYA
6CB4 48         7460     PHA
6CB5 B9C36C     7480     LDA RFLPAT,Y
6CB8 A8         7500     TAY
                7520 ;
6CB9 A5D8       7540     LDA R+4
6CBB 91DA       7560     STA (R+6),Y
                7580 ;
6CBD 68         7600     PLA
6CBE A8         7620     TAY
6CBF 88         7640     DEY
6CC0 10D1       7660     BPL REFL10
                7680 ;
6CC2 60         7700     RTS
                7720 ;
                7740 ; Reflected byte pattern
6CC3            7760 RFLPAT
6CC3 03020100   8081     .BYTE 3,2,1,0,4
```

Moore Business Forms, Inc. sv 1412RH

```
                        7740 ; Reflected byte pattern
6CC3                    7760 RFLPAT
6CC3 03020100           8081      .BYTE 3,2,1,0,4
```



```
6CC7 04
6CC8 08070605           8082      .BYTE 8,7,6,5,9
6CCC 09
6CCD 0D0C0B0A           8083      .BYTE 13,12,11,10,14
6CD1 0E
6CD2 1211100F           8084      .BYTE 18,17,16,15,19
6CD6 13
6CD7 17161514           8085      .BYTE 23,22,21,20,24
6CDB 18
6CDC 1C1B1A19           8086      .BYTE 28,27,26,25,29
6CE0 1D
6CE1 21201F1E           8087      .BYTE 33,32,31,30,34
6CE5 22
6CE6 26252423           8088      .BYTE 38,37,36,35,39
6CEA 27
6CEB 2B2A2928           8089      .BYTE 43,42,41,40,44
6CEF 2C
6CF0 302F2E2D           8090      .BYTE 48,47,46,45,49
6CF4 31
6CF5 35343332           8091      .BYTE 53,52,51,50,54
6CF9 36
6CFA 3A393837           8092      .BYTE 58,57,56,55,59
6CFE 3B
6CFF 3F3E3D3C           8093      .BYTE 63,62,61,60,64
6D03 40
6D04 44434241           8094      .BYTE 68,67,66,65,69
6D08 45
6D09 49484746           8095      .BYTE 73,72,71,70,74
6D0D 4A
6D0E 4E4D4C4B           8096      .BYTE 78,77,76,75,79
6D12 4F
                        8100 ;
                        8120 ;
                        8140 ; BACKSAVE DATA ADDRESS TABLE
                        8160 ;
6D13                    8180 BSTBLL ;         low
6D13 2373C313           8200      .BYTE  <BAKSAV, <BAKSAV+80, <BAKSAV+160, <BAKSAV+240
6D17 63B30353           8220      .BYTE  <BAKSAV+320, <BAKSAV+400, <BAKSAV+480,
                                           <BAKSAV+560
6D1B                    8240 BSTBLH ;         high
6D1B 6D6D6D6E           8260      .BYTE  >BAKSAV, >BAKSAV+80, >BAKSAV+160, >BAKSAV+240
6D1F 6E6E6F6F           8280      .BYTE  >BAKSAV+320, >BAKSAV+400, >BAKSAV+480,
                                           >BAKSAV+560
                        8300 ;
                        8320 ; BACKSAVE DATA AREA
                        8340 ;
6D23                    8360 BAKSAV *= *+OBJMAX*80
                        015860    .INCLUDE #D2:TRUCKS.M65
                        1000 ; TRUCK DATA  POS=0
6DC3                    1005 TRUCK
6DC3 00000000           1010      .BYTE $00,$00,$00,$00,$00
6DC7 00
6DC8 00000000           1020      .BYTE $00,$00,$00,$00,$00
6DCC 00
6DCD 00055400           1030      .BYTE $00,$05,$54,$00,$00
6DD1 00
6DD2 00155500           1040      .BYTE $00,$15,$55,$00,$00
```

Moore Business Forms, Inc. sv 1472RH

```
6DD6 00
6DD7 005FFD00   1050    .BYTE $00,$5F,$FD,$00,$00
6DDB 00
6DDC 15DFFD00   1060    .BYTE $15,$DF,$FD,$00,$00
6DE0 00
6DE1 55D55540   1070    .BYTE $55,$D5,$55,$40,$00
6DE5 00
6DE6 955FFF54   1080    .BYTE $95,$5F,$FF,$54,$00
6DEA 00
6DEB 9557FFF5   1090    .BYTE $95,$57,$FF,$F5,$00
6DEF 00
6DF0 9555FFFD   1100    .BYTE $95,$55,$FF,$FD,$00
6DF4 00
6DF5 1A555555   1110    .BYTE $1A,$55,$55,$55,$00
6DF9 00
6DFA 2A955555   1120    .BYTE $2A,$95,$55,$55,$00
6DFE 00
6DFF 00AAAAAA   1130    .BYTE $00,$AA,$AA,$AA,$00
6E03 00
6E04 00A56A95   1140    .BYTE $00,$A5,$6A,$95,$00
6E08 00
6E09 00AA95AA   1150    .BYTE $00,$AA,$95,$AA,$00
6E0D 00
6E0E 002A002A   1160    .BYTE $00,$2A,$00,$2A,$00
6E12 00
                1170 ; POS=1
6E13 00015500   1180    .BYTE $00,$01,$55,$00,$00
6E17 00
6E18 00055540   1190    .BYTE $00,$05,$55,$40,$00
6E1C 00
6E1D 0017FF40   1200    .BYTE $00,$17,$FF,$40,$00
6E21 00
6E22 0077FF40   1210    .BYTE $00,$77,$FF,$40,$00
6E26 00
6E27 05755540   1220    .BYTE $05,$75,$55,$40,$00
6E2B 00
6E2C 155FFF50   1230    .BYTE $15,$5F,$FF,$50,$00
6E30 00
6E31 255FFF54   1240    .BYTE $25,$5F,$FF,$54,$00
6E35 00
6E36 2557FFD4   1250    .BYTE $25,$57,$FF,$D4,$00
6E3A 00
6E3B 2957FFD5   1260    .BYTE $29,$57,$FF,$D5,$00
6E3F 00
6E40 2955FFF5   1270    .BYTE $29,$55,$FF,$F5,$00
6E44 00
6E45 0055FFF5   1280    .BYTE $00,$55,$FF,$F5,$00
6E49 00
6E4A 00555555   1290    .BYTE $00,$55,$55,$55,$00
6E4E 00
6E4F 00555555   1300    .BYTE $00,$55,$55,$55,$00
6E53 00
6E54 00AAAAAA   1310    .BYTE $00,$AA,$AA,$AA,$00
6E58 00
6E59 00AA80AA   1320    .BYTE $00,$AA,$80,$AA,$00
6E5D 00
6E5E 002A002A   1330    .BYTE $00,$2A,$00,$2A,$00
6E62 00
                1350 ; TRUCK DATA  POS=0
```

Moore Business Forms, Inc. sv 1472RH

```
6E5D 00
6E5E 002A002A    1330      .BYTE $00,$2A,$00,$2A,$00
6E62 00
                 1350 ; TRUCK DATA  POS=0
```



```
6E63             1355 TRUCKM
6E63 FFFFFFFF    1360      .BYTE $FF,$FF,$FF,$FF,$FF
6E67 FF
6E68 FFFFFFFF    1370      .BYTE $FF,$FF,$FF,$FF,$FF
6E6C FF
6E6D FFF003FF    1380      .BYTE $FF,$F0,$03,$FF,$FF
6E71 FF
6E72 FFC000FF    1390      .BYTE $FF,$C0,$00,$FF,$FF
6E76 FF
6E77 FF0000FF    1400      .BYTE $FF,$00,$00,$FF,$FF
6E7B FF
6E7C C00000FF    1410      .BYTE $C0,$00,$00,$FF,$FF
6E80 FF
6E81 0000003F    1420      .BYTE $00,$00,$00,$3F,$FF
6E85 FF
6E86 00000003    1430      .BYTE $00,$00,$00,$03,$FF
6E8A FF
6E8B 00000000    1440      .BYTE $00,$00,$00,$00,$FF
6E8F FF
6E90 00000000    1450      .BYTE $00,$00,$00,$00,$FF
6E94 FF
6E95 C0000000    1460      .BYTE $C0,$00,$00,$00,$FF
6E99 FF
6E9A C0000000    1470      .BYTE $C0,$00,$00,$00,$FF
6E9E FF
6E9F FF000000    1480      .BYTE $FF,$00,$00,$00,$FF
6EA3 FF
6EA4 FF000000    1490      .BYTE $FF,$00,$00,$00,$FF
6EA8 FF
6EA9 FF000000    1500      .BYTE $FF,$00,$00,$00,$FF
6EAD FF
6EAE FFC0FFC0    1510      .BYTE $FF,$C0,$FF,$C0,$FF
6EB2 FF
                 1520 ; POS=1
6EB3 FFFC00FF    1530      .BYTE $FF,$FC,$00,$FF,$FF
6EB7 FF
6EB8 FFF003F     1540      .BYTE $FF,$F0,$00,$3F,$FF
6EBC FF
6EBD FFC0003F    1550      .BYTE $FF,$C0,$00,$3F,$FF
6EC1 FF
6EC2 FF00003F    1560      .BYTE $FF,$00,$00,$3F,$FF
6EC6 FF
6EC7 F0000003F   1570      .BYTE $F0,$00,$00,$3F,$FF
6ECB FF
6ECC C000000F    1580      .BYTE $C0,$00,$00,$0F,$FF
6ED0 FF
6ED1 C0000003    1590      .BYTE $C0,$00,$00,$03,$FF
6ED5 FF
6ED6 C0000003    1600      .BYTE $C0,$00,$00,$03,$FF
6EDA FF
6EDB C0000000    1610      .BYTE $C0,$00,$00,$00,$FF
6EDF FF
6EE0 C0000000    1620      .BYTE $C0,$00,$00,$00,$FF
6EE4 FF
6EE5 FF000000    1630      .BYTE $FF,$00,$00,$00,$FF
6EE9 FF
6EEA FF000000    1640      .BYTE $FF,$00,$00,$00,$FF
```

Moore Business Forms, Inc. sv 1412RH

```
6EEE FF

6EEF FF000000   1650      .BYTE $FF,$00,$00,$00,$FF
6EF3 FF
6EF4 FF000000   1660      .BYTE $FF,$00,$00,$00,$FF
6EF8 FF
6EF9 FF003F00   1670      .BYTE $FF,$00,$3F,$00,$FF
6EFD FF
6EFE FFC0FFC0   1680      .BYTE $FF,$C0,$FF,$C0,$FF
6F02 FF
                1690 ;  END OF TRUCK DATA
                1700 ;
6F03            1710 SWEEPR
6F03            1720 SWEEPRM
6F03            1730 FLGCAR
6F03            1740 FLGCARM
                1750 ;
                1760 ;
                015880    .INCLUDE #D1:FLAGS.M65
                1000      .PAGE "FLAGS.M65"
```

Moore Business Forms, Inc. sv 14128M

```
                    1020 ;
                    1040 ; FLAGS.M65
                    1060 ; FLAG ROUTINES
                    1080 ;
6F03                1100 FLGCNT *= *+1    ; count of flags left
6F04                1120 FTOGL *= *+1     ; enable flag toggles
6F05                1140 FHIT *= *+1      ; last flag hit or currently hit
6F06                1160 FDLYCT *= *+1    ; flag delay ctr
     =0050          1180 FDLY =  80       ; delay betw flag hit checks
     =D00B          1200 M3PL =  $D00B    ; missile to player hit
     =D01E          1220 HITCLR = $D01E   ; clear hits write location
                    1240 ;
                    1260 ; *FLGHIT*
                    1280 ; HANDLES FLAG TOGGLING
                    1300 ;
6F07                1320 FLGHIT
6F07 CE066F         1340     DEC FDLYCT
6F0A F003           1360     BEQ FLGH04
6F0C 4C856F         1380     JMP FLGH90
6F0F                1400 FLGH04
6F0F A950           1420     LDA #FDLY    ; reset delay
6F11 8D066F         1440     STA FDLYCT
6F14 AD0BD0         1460     LDA M3PL     ; any hits?
6F17 D003           1480     BNE FLGH06
6F19 4C806F         1500     JMP FLGH80   ; no - branch
6F1C                1520 FLGH06
6F1C 8D1ED0         1540     STA HITCLR   ; clear hits
                    1560 ;
                    1580 ; Find out which flag is hit
                    1600 ;
6F1F A209           1620     LDX #9
6F21                1640 FLGH10
6F21 BDA46F         1660     LDA FYH,X    ; can't have MSB of Y <> 0
6F24 D028           1680     BNE FLGH40
6F26 BD9A6F         1700     LDA FYL,X    ; check Y
6F29 38             1720     SEC
6F2A E904           1740     SBC #4
6F2C EDA942         1760     SBC BUGY
6F2F 1005           1780     BPL FLGH20
6F31 49FF           1800     EOR #$FF
6F33 18             1820     CLC
6F34 6901           1840     ADC #1
6F36                1860 FLGH20
6F36 C909           1880     CMP #9
6F38 B014           1900     BCS FLGH40
6F3A BDB86F         1920     LDA FHPOS,X  ; check x
6F3D 38             1940     SEC          ; convert from P/M
6F3E E934           1960     SBC #52      ; won't ovfl
6F40 EDA842         1980     SBC BUGX
6F43 1005           2000     BPL FLGH30
6F45 49FF           2020     EOR #$FF
6F47 18             2040     CLC
6F48 6901           2060     ADC #1
6F4A                2080 FLGH30
6F4A C909           2100     CMP #9
6F4C 9006           2120     BCC FLGH50
6F4E                2140 FLGH40
```

```
6F4E CA         2160     DEX          ; next flag



6F4F 10D0       2180     BPL FLGH10
6F51 4C856F     2200     JMP FLGH90  ; error exit
                2220 ;
                2240 ; Got flag
                2260 ;
6F54            2280 FLGH50
6F54 EC056F     2300     CPX FHIT    ; is this hit a continuation of last?
6F57 F02C       2320     BEQ FLGH90  ; yes - branch
6F59 8E056F     2340     STX FHIT    ; save currnet flag hit
6F5C BDAE6F     2360     LDA FCOLOR,X ; is flag on or off?
6F5F 2901       2380     AND #1
6F61 F00B       2400     BEQ FLGH60  ; on - branch
6F63 AD046F     2420     LDA FTOGL   ; toggle enabled?
6F66 F01D       2440     BEQ FLGH90  ; no - branch
6F68 EE036F     2460     INC FLGCNT
6F6B 4C716F     2480     JMP FLGH61
6F6E            2500 FLGH60
6F6E CE036F     2520     DEC FLGCNT
6F71            2540 FLGH61
6F71 207B66     2560     JSR FLAG    ; do flag sound
6F74 BDAE6F     2580     LDA FCOLOR,X ; change flag
6F77 4901       2600     EOR #1
6F79 9DAE6F     2620     STA FCOLOR,X
                2640 ;
6F7C 20E571     2660     JSR CHGFLG  ; change header flag
6F7F 60         2680     RTS
                2700 ;
6F80            2720 FLGH80
6F80 A9FF       2740     LDA #$FF
6F82 8D056F     2760     STA FHIT
6F85            2780 FLGH90
6F85 60         2800     RTS
                2820 ;
                2840 ;
6F86            2860 IFYL ;          flag Ypos high byte
6F86 C797E787   2880     .BYTE  <455, <407, <743, <647, <615, <847, <791, <327,
                                  <151, <119
6F8A 674F1747
6F8E 9777
6F90            2900 IFYH ;          "      low    "
6F90 01010202   2920     .BYTE  >455, >407, >743, >647, >615, >847, >791, >327,
                                  >151, >119
6F94 02030301
6F98 0000
6F9A            2940 FYL *=  *+10
6FA4            2960 FYH *=  *+10
6FAE            2980 FCOLOR ;        flag color byte
6FAE 164A3656   3000     .BYTE $16,$4A,$36,$56,$AA
6FB2 AA
6FB3 861E861E   3020     .BYTE $86,$1E,$86,$1E,$86
6FB7 86
6FB8            3040 FHPOS ;         (#PFLINES=983)flag horiz. pos.
6FB8 689888A8   3060     .BYTE 104,152,136,168,98
6FBC 62
6FBD A658B546   3080     .BYTE 166,88,181,70,168
6FC1 A8
                3100 ;
6FC2            3120 IHFPFL ;        PF flags horiz pos, low
```

Moore Business Forms, Inc. sv 14128H

```
6FBB                3040 FIPOS ;          (6502_Intro-Sec)flag horiz. pos.
6FBB 5A5B5C5D       3060      .BYTE 104,105,106,107,92
6FBC 62
6FBD A658B546       3080      .BYTE 166,88,181,70,168
6FC1 A8
                    3100 ;
6FC2                3120 IHFPFL ;         PF flags horiz pos, low
```



```
6FC2 1014181C       3140      .BYTE  <HMEM+90, <HMEM+90+4, <HMEM+90+8, <HMEM+90+12,
                                     <HMEM+90+16
6FC6 20
6FC7 12161A1E       3160      .BYTE  <HMEM+90+2, <HMEM+90+6, <HMEM+90+10,
                                     <HMEM+90+14, <HMEM+90+18
6FCB 22
6FCC                3180 IHFPFH ;         high
6FCC 48484848       3200      .BYTE  >HMEM+90, >HMEM+90+4, >HMEM+90+8, >HMEM+90+12,
                                     >HMEM+90+16
6FD0 48
6FD1 48484848       3220      .BYTE  >HMEM+90+2, >HMEM+90+6, >HMEM+90+10,
                                     >HMEM+90+14, >HMEM+90+18
6FD5 48
6FD6                3240 IHFPFC ;         playfield flag color
6FD6 55555555       3260      .BYTE $55,$55,$55,$55,$55
6FDA 55
6FDB AAFFAAFF       3280      .BYTE $AA,$FF,$AA,$FF,$AA
6FDF AA
                    3300 ;
                    3320 ;
                    3340 ; INIT THE FLAGS
                    3360 ;
6FE0                3380 IFLAGS
6FE0 A209           3400      LDX #9
6FE2                3420 IFLA10
6FE2 BD866F         3440      LDA IFYL,X
6FE5 9D9A6F         3460      STA FYL,X
6FE8 BD906F         3480      LDA IFYH,X
6FEB 9DA46F         3500      STA FYH,X
6FEE BDAE6F         3510      LDA FCOLOR,X
6FF1 29FE           3512      AND #$FE
6FF3 9DAE6F         3514      STA FCOLOR,X
6FF6 CA             3520      DEX
6FF7 10E9           3540      BPL IFLA10
                    3560 ;
6FF9 20FD6F         3580      JSR IHFLGS
6FFC 60             3600      RTS
                    3620 ;
                    3640 ; SET UP THE FLAGS IN THE HEADER
                    3660 ;
6FFD                3680 IHFLGS
6FFD A20F           3700      LDX #15     ; init P/M data
6FFF                3720 IHFL10
6FFF BD9271         3740      LDA MFLGDAT+2,X
7002 9D3273         3760      STA M0+50,X
7005 9D3274         3780      STA P0+50,X
7008 9D3275         3800      STA P1+50,X
700B 9D3276         3820      STA P2+50,X
700E 9D3277         3840      STA P3+50,X
7011 CA             3860      DEX
7012 10EB           3880      BPL IHFL10
                    3900 ;
7014 A204           3920      LDX #4      ; init HPOS data
7016                3940 IHFL20
7016 BD3770         3960      LDA IHFPM,X
7019 9D3C70         3980      STA HFPOS,X
701C CA             4000      DEX
```

Moore Business Forms, Inc. sv 1412RH

```
                    4020      BPL IHFL20



                    4040 ;
701F A209           4060      LDX #9      ; init 10 header flags
7021                4080 IHFL30
7021 BDC26F         4100      LDA IHFPFL,X
7024 85D4           4120      STA R+0
7026 BDCC6F         4140      LDA IHFPFH,X
7029 85D5           4160      STA R+1
702B BDD66F         4180      LDA IHFPFC,X
702E 85D7           4200      STA R+3
7030 20AB54         4220      JSR EFLAG   ; draw flag
7033 CA             4240      DEX
7034 10EB           4260      BPL IHFL30
                    4280 ;
7036 60             4300      RTS
                    4320 ;
                    4340 ; Initial header flag horiz pos.
                    4360 ; For P/M flags
7037                4380 IHFPM
7037 58687888       4400      .BYTE 88,104,120,136,152
703B 98
703C                4420 HFPOS *= *+5     ; horiz pos of header P/M flags
                    4440 ;
                    4460 ;
                    4480 ; Scroll the flags in the flag tables
                    4500 ;
7041                4520 FLGSCR
7041 ADDB5A         4540      LDA SCRFLG  ; only do this after screen scroll
7044 F043           4560      BEQ FLGS45
                    4580 ;
7046 A96E           4600      LDA #110
7048                4620 FLGS01
7048 CD0BD4         4640      CMP VCOUNT  ; sync to screen
704B B0FB           4660      BCS FLGS01
                    4680 ;
                    4700 ;
                    4720 ; Setup for RSTDLI routine in
                    4740 ; case it's called
                    4760 ;
704D A900           4780      LDA #0      ; unflag 1st DLI
704F 8DE471         4800      STA FTOPDLI
                    4820 ;
7052 A9FF           4840      LDA #$FF    ; init list w/large #
7054 A204           4860      LDX #MAXFON
7056                4880 RSTD08
7056 9DDE71         4900      STA FDLISEQ,X
7059 CA             4920      DEX
705A 10FA           4940      BPL RSTD08
                    4960 ;
705C ADDB5A         4980      LDA SCRFLG  ; which way was scroll?
705F 3029           5000      BMI FLGS50
                    5020 ;
                    5040 ; Scroll flag data down
                    5060 ;
7061 A209           5080      LDX #9      ; index to flags
7063                5100 FLGS10
7063 FE9A6F         5120      INC FYL,X   ; chg flag's Y
7066 D003           5140      BNE FLGS15
7068 FEA46F         5160      INC FYH,X


```

Moore Business Forms, Inc. sv 1412RH

```
7063 FE9A6F     5120     INC FYL,X   ; chg flag's Y
7066 D003       5140     BNE FLGS15
7068 FEA46F     5160     INC FYH,X
```



```
                5180 FLGS15
706B BDA46F     5200     LDA FYH,X   ; check for wrap
706E C903       5220     CMP # >APFLEN/10Ü*16
7070 9011       5240     BCC FLGS30
7072 D007       5260     BNE FLGS20
7074 BD9A6F     5280     LDA FYL,X
7077 C9E0       5300     CMP # <APFLEN/10Ü*16
7079 9008       5320     BCC FLGS30
707B            5340 FLGS20
707B A900       5360     LDA #0      ; do wraparound
707D 9D9A6F     5380     STA FYL,X
7080 9DA46F     5400     STA FYH,X
7083            5420 FLGS30
7083 20B170     5440     JSR DRWFLG
7086            5460 FLGS40
7086 CA         5480     DEX
7087 10DA       5500     BPL FLGS10
7089            5520 FLGS45
7089 60         5540     RTS
                5560 ;
                5580 ; Scroll flag data up
                5600 ;
708A            5620 FLGS50
708A A209       5640     LDX #9
708C            5660 FLGS55
708C DE9A6F     5680     DEC FYL,X
708F BD9A6F     5700     LDA FYL,X
7092 C9FF       5720     CMP #$FF
7094 D003       5740     BNE FLGS60
7096 DEA46F     5760     DEC FYH,X
7099            5780 FLGS60
7099 BDA46F     5800     LDA FYH,X   ; check for wrap
709C C9FF       5820     CMP #$FF
709E D00A       5840     BNE FLGS70
70A0 A9DF       5860     LDA # <APFLEN/10Ü*16-1 ; do wraparound
70A2 9D9A6F     5880     STA FYL,X
70A5 A903       5900     LDA # >APFLEN/10Ü*16-1
70A7 9DA46F     5920     STA FYH,X
70AA            5940 FLGS70
70AA 20B170     5960     JSR DRWFLG
70AD            5980 FLGS80
70AD CA         6000     DEX
70AE 10DC       6020     BPL FLGS55
70B0 60         6040     RTS
                6060 ;
                6080 ;
                6100 ; Draw the flag as a 5th player
                6120 ; by combining all missiles
                6140 ; uses R+10,11,12,13.  Set DLI
                6160 ;
70B1            6180 DRWFLG
70B1 8A         6200     TXA
70B2 48         6220     PHA
70B3 BDA46F     6240     LDA FYH,X   ; check within display bounds
70B6 D070       6260     BNE DRWF50
70B8 BD9A6F     6280     LDA FYL,X
70BB C90B       6300     CMP #11
```

Moore Business Forms, Inc. sv 1412RH

```
70BF C963       6340    CMP #99
70C1 B065       6360    BCS DRWF50
                6380 ;
70C3 85DF       6400    STA R+11    ; save flag's Y
70C5 86E0       6420    STX R+12    ; save flag #
70C7 C913       6440    CMP #19     ; clip top?
70C9 9036       6460    BCC DRWF30  ; yes - branch
70CB C95C       6480    CMP #92     ; clip bottom?
70CD B048       6500    BCS DRWF40  ; yes - branch
                6520 ;
                6540 ;
70CF A912       6560    LDA #18     ; # bytes to draw
70D1 85DE       6580    STA R+10
70D3 BD9A6F     6600    LDA FYL,X   ; set index to dest.
70D6 A200       6620    LDX #0      ; index to source data
70D8            6640 DRWF05
70D8 0A         6660    ASL A       ; convert to P/M coords
70D9 18         6680    CLC
70DA 691D       6700    ADC #29
70DC A8         6720    TAY
70DD            6740 DRWF20
70DD 98         6760    TYA         ; sync to screen
70DE 4A         6780    LSR A
70DF 38         6800    SEC
70E0 E903       6820    SBC #3
70E2 CD0BD4     6840    CMP VCOUNT
70E5 B008       6860    BCS DRWF22
70E7 18         6880    CLC
70E8 6906       6900    ADC #6
70EA            6920 DRWF21
70EA CD0BD4     6940    CMP VCOUNT
70ED B0FB       6960    BCS DRWF21
70EF            6980 DRWF22
70EF BD9071     7000    LDA MFLGDAT,X ; draw flag
70F2 990073     7020    STA MO,Y
70F5 E8         7040    INX
70F6 C8         7060    INY
70F7 C6DE       7080    DEC R+10
70F9 D0F4       7100    BNE DRWF22
                7120 ;
70FB 202B71     7140    JSR RSTDLI  ; set DLI
                7160 ;
70FE 4C2871     7180    JMP DRWF50
                7200 ;
                7220 ; Clip top
                7240 ;
7101            7260 DRWF30
7101 A913       7280    LDA #19
7103 38         7300    SEC
7104 FD9A6F     7320    SBC FYL,X
7107 0A         7340    ASL A
7108 85DE       7360    STA R+10
710A AA         7380    TAX
710B A912       7400    LDA #18
710D 38         7420    SEC
710E E5DE       7440    SBC R+10
7110 85DE       7460    STA R+10
7112 A043       7480    LDY #67
```

Moore Business Forms, Inc. sv 1412RH

```
710A AA          7380      TAX
710B A912        7400      LDA #18
710D 38          7420      SEC
710E E5DE        7440      SBC R+10
7110 85DE        7460      STA R+10
7112 A043        7480      LDY #67
```



```
7114 4CDD70      7500      JMP DRWF20
                 7520 ;
                 7540 ; Clip bottom
                 7560 ;
7117             7580 DRWF40
7117 A963        7600      LDA #99
7119 38          7620      SEC
711A FD9A6F      7640      SBC FYL,X
711D 0A          7660      ASL A
711E 85DE        7680      STA R+10
7120 BD9A6F      7700      LDA FYL,X
7123 A200        7720      LDX #0
7125 4CD870      7740      JMP DRWF05
7128             7760 DRWF50
7128 68          7780      PLA
7129 AA          7800      TAX
712A 60          7820      RTS
                 7840 ;
                 7860 ;
                 7880 ; Reset flag DLI's to control
                 7900 ; flag's color and HPOS
                 7920 ; R+11    flag's Y
                 7940 ; R+12    flag #
                 7960 ; R+13    used
712B             7980 RSTDLI
                 8000 ;
712B A5DF        8020      LDA R+11    ; compute index to DLIST
712D 38          8040      SEC         ; based on flag's Y
712E E913        8060      SBC #19
7130 A8          8080      TAY
7131 B93C4B      8100      LDA M3TBL,Y ; *3
7134 A8          8120      TAY
7135 A5DF        8140      LDA R+11    ; top line or above?
7137 C914        8160      CMP #20
7139 9005        8180      BCC RSTD10  ; yes - branch
713B A96D        8200      LDA #$6D    ; clear top adjacent DLI
713D 99C042      8220      STA DLIST+25-3,Y
7140             8240 RSTD10
7140 A5DF        8260      LDA R+11    ; top clip?
7142 C913        8280      CMP #19
7144 B00D        8300      BCS RSTD30  ; no - branch
7146 A901        8320      LDA #1      ; set top DLI flag
7148 8DE471      8340      STA FTOPDLI
714B A96D        8360      LDA #$6D    ; erase last DLI
714D 8DC342      8380      STA DLIST+25
7150 4C6371      8400      JMP RSTD32
7153             8420 RSTD30
7153 A5DF        8440      LDA R+11    ; bottom line or below?
7155 C962        8460      CMP #98
7157 B005        8480      BCS RSTD31  ; yes - branch
7159 A96D        8500      LDA #$6D    ; clear bottom adjacent DLI
715B 99C642      8520      STA DLIST+25+3,Y
715E             8540 RSTD31
715E A9ED        8560      LDA #$ED    ; set DLI
7160 99C342      8580      STA DLIST+25,Y
7163             8600 RSTD32
                 8620 ;
```

Moore Business Forms, Inc. sv 14128H

```
                        8640 ; Put flag in FDLISEQ list in order
                        8660 ; of display
                        8680 ;
7163 A000               8700     LDY #0
7165                    8720 RSTD35
7165 B9DE71             8740     LDA FDLISEQ,Y ; find a flag
7168 C90A               8760     CMP #10       ; valid flag #?
716A B008               8780     BCS RSTD40    ; no - branch
716C AA                 8800     TAX           ; else get this flag's onscreen Y
716D BD9A6F             8820     LDA FYL,X
7170 C5DF               8840     CMP R+11      ; current flag displayed above this?
7172 9012               8860     BCC RSTD48    ; no - branch
7174                    8880 RSTD40
7174 84E1               8900     STY R+13      ; current flag goes in this slot
7176 A004               8920     LDY #MAXFON   ; max # flags on screen at once
7178                    8940 RSTD44
7178 B9DE71             8960     LDA FDLISEQ,Y
717B 99DF71             8980     STA FDLISEQ+1,Y
717E C4E1               9000     CPY R+13
7180 F008               9020     BEQ RSTD50
7182 88                 9040     DEY
7183 4C7871             9060     JMP RSTD44
7186                    9080 RSTD48
7186 C8                 9100     INY
7187 4C6571             9120     JMP RSTD35
                        9140 ;
718A                    9160 RSTD50
718A A5E0               9180     LDA R+12
718C 99DE71             9200     STA FDLISEQ,Y ; put flag in slot
718F 60                 9220     RTS
                        9240 ;
                        9260 ;
                        9280 ;
                        9300 ; MISSILE DATA FOR FLAGS
                        9320 ;
7190                    9340 MFLGDAT
7190 00007078           9360     .BYTE 0,0,$70,$78,$7C,$7E,$7E,$7C,$78,$70,$40,$40,$40,
                                      $40,$40,$40,0,0
7194 7C7E7E7C
7198 78704040
719C 40404040
71A0 0000
                        9380 ;
                        9400 ;
                        9420 ; FLAG DLI ROUTINE SETS COLOR
                        9440 ; AND HORIZONTAL POSITION OF
                        9460 ; FLAGS.
                        9480 ; FLAGS IN "FDLISEQ" TABLE ARE
                        9500 ; HANDLED IN ORDER AND INDEXED
                        9520 ; WITH "FDLINDX"
                        9540 ;
71A2                    9560 FDLI
71A2 48                 9580     PHA
71A3 98                 9600     TYA
71A4 48                 9620     PHA
71A5 8A                 9640     TXA
71A6 48                 9660     PHA
                        9680 ;
71A7 AEE371             9700     LDX FDLINDX ; get index to next flag.
```

```
71A4 48          9620    PHA
71A5 8A          9640    TXA
71A6 48          9660    PHA
                 9680 ;
71A7 AEE371      9700    LDX FDLINDX ; get index to next flag
```



```
71AA BDDE71      9720    LDA FDLISEQ,X ; get next flag #
71AD AA          9740    TAX
71AE BDAE6F      9760    LDA FCOLOR,X ; set color
71B1 2901        9780    AND #1       ; use color or white?
71B3 F005        9800    BEQ FDLI10   ; use color - branch
71B5 A90F        9820    LDA #$0F     ; use white
71B7 4CBD71      9840    JMP FDLI20
71BA             9860 FDLI10
71BA BDAE6F      9880    LDA FCOLOR,X
71BD             9900 FDLI20
71BD 8D19D0      9920    STA $D019
                 9940 ;
71C0 BCB86F      9960    LDY FHPOS,X ; set horiz pos.
71C3 8C07D0      9980    STY HPOSM3
71C6 C8          010000  INY
71C7 C8          010020  INY
71C8 8C06D0      010040  STY HPOSM2
71CB C8          010060  INY
71CC C8          010080  INY
71CD 8C05D0      010100  STY HPOSM1
71D0 C8          010120  INY
71D1 C8          010140  INY
71D2 8C04D0      010160  STY HPOSM0
                 010180 ;
                 010200 ;
71D5 EEE371      010220  INC FDLINDX ; inc index for next DLI / next flag
                 010240 ;
71D8 68          010260  PLA
71D9 AA          010280  TAX
71DA 68          010300  PLA
71DB A8          010320  TAY
71DC 68          010340  PLA
71DD 40          010360  RTI
                 010380 ;
                 010400 ;
                 010420 ; FDLISEQ table holds order of
                 010440 ; flags on screen from top to
                 010460 ; bottom for DLI's to access.
                 010480 ; Max number of flags on screen
                 010500 ; is 8.
                 010520 ;
    =0004        010540 MAXFON = 4   ; max # of flags on screen at one time
71DE             010560 FDLISEQ *= *+MAXFON+1
71E3             010580 FDLINDX *= *+1 ; index to FDLISEQ
                 010600 ;
                 010620 ;
                 010640 ;
71E4             010660 FTOPDLI *= *+1 ; flag for 1st DLI
                 010680 ;
                 010700 ;
                 010720 ;
                 010740 ;
                 010760 ; Change flag header display
                 010780 ;
71E5             010800 CHGFLG
71E5 E005        010820   CPX #5      ; is header flag P/M or PF?
71E7 9021        010840   BCC CHGFZ0  ; P/M - branch
```

Moore Business Forms, Inc. sv 1412MM

```
                        010860 ;
                        010880 ; Flag is Playfield flag
                        010900 ;
 71E9 BDC26F            010920   LDA IHFPFL,X ; set dest ptr
 71EC 85D4              010940   STA R+0
 71EE BDCC6F            010960   LDA IHFPFH,X
 71F1 85D5              010980   STA R+1
 71F3 BDAE6F            011000   LDA FCOLOR,X ; which color?
 71F6 2901              011020   AND #1
 71F8 F005              011040   BEQ CHGF62
 71FA A955              011060   LDA #$55    ; make white
 71FC 4C0272            011080   JMP CHGF64
 71FF                   011100 CHGF62
 71FF BDD66F            011120   LDA IHFPFC,X ; make color
 7202                   011140 CHGF64
 7202 85D7              011160   STA R+3
 7204 20AB54            011180   JSR EFLAG   ; draw flag
 7207 4C1C72            011200   JMP CHGF90
                        011220 ;
                        011240 ; Flag is a P/M flag
                        011260 ;
 720A                   011280 CHGF70
 720A BDAE6F            011300   LDA FCOLOR,X
 720D 2901              011320   AND #1
 720F F005              011340   BEQ CHGF74
 7211 A900              011360   LDA #0      ; move flag off screen to reveal white
 7213 4C1972            011380   JMP CHGF76
 7216                   011400 CHGF74
 7216 BD3770            011420   LDA IHFPM,X ; move back on screen
 7219                   011440 CHGF76
 7219 9D3C70            011460   STA HFPOS,X
                        011480 ;
 721C                   011500 CHGF90
 721C 60                011520   RTS
                        015900 ;
                        015920 ;
    =721D               015960 PROGEND = *
 721D                   016000   .END

ASSEMBLY ERRORS: 0   18118 BYTES FREE
```

Moore Business Forms, Inc. sv 1412RH

```
 5ADF ACCCNT        =0028 ACCDLY       5FBB ADDP1        5353 ADDS10
 5357 ADDS18        5375 ADDS20        537E ADDS24       533C ADDSCOR
 66BC ALLFLGS       =004D ATRACT       =D201 AUDC1       =D208 AUDCTL
=D200 AUDF1         68FD AUDV1         =68FF AUDV2       =6901 AUDV3
=6903 AUDV4
 531A B2DL05        5326 B2DL10        5338 B2DL20       6D23 BAKSAV
 51DC BDATH         51D6 BDATL         52F2 BDLI1        530C BDLI2
 529E BINC          5AD7 BLSTINC       5AD8 BLSTNZR      51D0 BMSGH
 51CA BMSGL         51B3 BPAG00        509E BPAG10       50BB BPAG20
 50AA BPAG22        50CA BPAG25        50DC BPAG30       5104 BPAG40
 510F BPAG50        5157 BPAG54        5160 BPAG60       5162 BPAG61
 5178 BPAG70        518B BPAG80        5198 BPAG90       51A4 BPAG92
 5099 BPAGE         505C BPAGEF        51FE BPDAT        5ADA BPTHCT
 6894 BR419         689B BR420         68CF BR421        68DA BR423
 68E2 BR424         68AE BR457         68B9 BR458        68BF BR459
 6891 BR461         6887 BR463         668B BSOUND       6D1B BSTBLH
 6D13 BSTBLL        5AD9 BUGDIR        5FE5 BUGH0        5FE6 BUGH1
 5FE7 BUGH2         5FE8 BUGH3         5FE9 BUGP0        60A9 BUGP1
 6169 BUGP2         6229 BUGP3         42A8 BUGX         42A9 BUGY
=FFF0 BWDSPD        5C21 BYTADD
 4296 CARLFT        5059 CFADE         =02FC CH          =02F2 CH1
 420C CHARH         424C CHARL         =9000 CHARS       =02F4 CHBAS
=D409 CHBASE        55A5 CHCOPY        4F7B CHGDSP       71FF CHGF62
 7202 CHGF64        720A CHGF70        7216 CHGF74       7219 CHGF76
 721C CHGF90        71E5 CHGFLG        4014 CHINDX       4B8C CHRREF
=0001 CINC          5058 CMSGCT        =0064 CMSGDY      505B CMSGON
 65D9 COLI10        65EC COLI20        65FE COLI30       6602 COLI40
 6607 COLI50        65D9 COLISN        =D012 COLPM0      =D013 COLPM1
=D014 COLPM2        =D015 COLPM3       505D CON          528A CONGRA
=D01F CONSOL        4015 CURCHAR       4010 CX           4011 CY
=3740 DATAX         6AA9 DATE05        6AB3 DATE10       6AC4 DATE15
 6AF8 DATE20        6B3E DATE30        6B57 DATE40       6AA5 DATEXP
 5A5D DESE10        5A6D DESE15        5A87 DESE20       5AD5 DESE30
 5AB2 DESE56        5AB4 DESE57        5A52 DESETUP      6307 DIR
 5C1B DIRING        53AA DLI1          53CE DLI2         542E DLI220
 42AA DLIST         =D400 DMACTL       6A03 DNSC01       6A10 DNSC02
 6A48 DNSC10        6A57 DNSC15        69D7 DNSCRN       59C0 DRAW
 59C2 DRAW03        59C9 DRAW04        59CF DRAW05       59D1 DRAW10
 59F7 DRAW15        5A05 DRAW20        5A06 DRAW30       634F DRAWN
 70D8 DRWF05        70DD DRWF20        70EA DRWF21       70EF DRWF22
 7101 DRWF30        7117 DRWF40        7128 DRWF50       70B1 DRWFLG
 400A DSCROL        411D DSPTBH        403E DSPTBL       6B79 DXPTR1H
 6B75 DXPTR1L       6B81 DXPTR2H       6B7D DXPTR2L
 5430 ECAR          543A ECAR10        543E ECAR20       5458 ECAR30
 5460 ECAR40        5467 ECAR50        547B ECAR60       54FA ECARDAT
 54B1 EFLA10        54D7 EFLA20        54AB EFLAG        =002D ENDJMP
 5064 ENDR10        506C ENDR20        5070 ENDR22       505E ENDRND
 5A19 ERAS04        5A1F ERAS05        5A14 ERAS06       5A21 ERAS10
 5A41 ERAS15        5A4F ERAS30        5A09 ERASE
 505A FADECT        6FAE FCOLOR        71A2 FDLI         71BA FDLI10
 71BD FDLI20        71E3 FDLINDX       71DE FDLISEQ      =0050 FDLY
 6F06 FDLYCT        6B5C FFBYTE        6F05 FHIT         6FB8 FHPOS
 667B FLAG          54DA FLAGDAT       5056 FLASH        6F03 FLGCAR
 6F03 FLGCARM       6F03 FLGCNT        6F0F FLGH04       6F1C FLGH06
 6F21 FLGH10        6F36 FLGH20        6F4A FLGH30       6F4E FLGH40
 6F54 FLGH50        6F6E FLGH60        6F71 FLGH61       6F80 FLGH80
```

6F85 FLGH90	6F07 FLGHIT	7048 FLGS01	7063 FLGS10
706B FLGS15	707B FLGS20	7083 FLGS30	7086 FLGS40
7089 FLGS45	708A FLGS50	708C FLGS55	7099 FLGS60
70AA FLGS70	70AD FLGS80	7041 FLGSCR	=00C8 FLSDLY
5057 FLSHCT	6F04 FTOGL	71E4 FTOPDLI	=0010 FWDSPD
6FA4 FYH	6F9A FYL		
=026F GPRIOR	=D01D GRACTL		
703C HFPOS	=D01E HITCLR	47B6 HMEM	=0370 HMEMSIZ
=D004 HPOSM0	=D005 HPOSM1	=D006 HPOSM2	=D007 HPOSM3
=D000 HPOSP0	=D001 HPOSP1	=D002 HPOSP2	=D003 HPOSP3
52C7 IBDL10	52A4 IBDLST	4F2B IDLI10	4F41 IDLI30
4F03 IDLIST	6FE2 IFLA10	6FE0 IFLAGS	6F90 IFYH
6F86 IFYL	6FFF IHFL10	7016 IHFL20	7021 IHFL30
6FFD IHFLGS	6FD6 IHFPFC	6FCC IHFPFH	6FC2 IHFPFL
7037 IHFPM	547E IHME10	5495 IHME20	54A2 IHME30
54A4 IHME40	547C IHMEM	5512 IHTXT	=0014 INCDLY
5055 INCTR	6444 IOBJ10	6440 IOBJTB	=7800 IPF
4CE0 IPFD10	4CF1 IPFD20	4CFA IPFD30	4CC6 IPFDATA
538F IPLY10	538B IPLYRDT	4C9B IPME10	4CA8 IPME20
4C88 IPMEM	6905 ISFX	6909 ISFX10	4EB5 ISME10
4EF8 ISME30	4EA4 ISMEM	4FA5 ITAB05	4FAD ITAB10
4FC7 ITAB15	4FD8 ITAB20	4FE7 ITAB28	4FEC ITAB30
4FF6 ITAB40	5011 ITAB50	5016 ITAB60	5023 ITAB70
4F99 ITABLES	556C ITCH10	5587 ITCH15	5589 ITCH20
559F ITCH30	5562 ITCHARS	5040 IVAR10	502C IVARS
4CFB IVBI			
5C3C JMPADD	5ADC JMPCTR	=000C JMPDLY	5AE0 JMPNDX
5C69 JMPPIC			
400E KX	400F KY		
6334 LASTINC	633D LASTNZR	42A7 LINCNT	=0004 LL
6872 LP405	=0008 LR		
=7300 M0	4B26 M10TBL	5F43 M32TBL	=D00B M3PL
4B3C M3TBL	41FC M4TBL	4B2C M5TBL	4D60 MAIN08
4D20 MAIN10	4D63 MAIN15	4D72 MAIN20	4D81 MAIN30
4D8D MAIN50	=2E80 MASKX	=0004 MAXFON	7190 MFLGDAT
6322 MODE	4F84 MOVD10	6358 MOVE	635A MOVE10
635F MOVE15	6363 MOVE16	636D MOVE20	6392 MOVE22
63B0 MOVE23	63B5 MOVE24	63C3 MOVE25	63C6 MOVE26
63D2 MOVE30	63D3 MOVE40	642A MOVE45	6435 MOVE50
=0002 NACTYPE	6609 NEWRND	=D40E NMIEN	=0008 NPATH
=0006 NTYPES			
6B85 OBJDAT	=0002 OBJMAX	6BC5 OBJMSK	6B5A OBJREF
5ADE OLDBDIR	5ADD OLDBUGY	4013 OLDHPOS	4012 OLDVPOS
62EB ONTOP	=4000 ORG	=E000 OSCHARS	6473 OVRL05
648A OVRL10	649C OVRL20	64A1 OVRL40	6471 OVRLAP
6346 OVRLAY	=0014 OVRLPX	=0014 OVRLPY	
=7400 P0	=7500 P1	=7600 P2	=7700 P3
=D302 PACTL	632B PATH	5C24 PATHD	=02C0 PCOLR0
=02C1 PCOLR1	=02C2 PCOLR2	=02C3 PCOLR3	43B6 PFDATA
=026C PFLEN	6A67 PFLI10	6A7F PFLI20	6A5B PFLINE
4297 PLYRDAT	=D407 PMBASE	=7000 PMEM	5D0C PMO330
5D1C PMO332	5D26 PMO333	5D2D PMO334	5D46 PMO342
5D5F PMO400	5D64 PMO405	5D6A PMO410	5DD7 PMO625
5DED PMO628	5CBA PMOV10	5CDC PMOV20	5CE4 PMOV25
5CE7 PMOV30	5CFC PMOV31	5CF4 PMOV32	5D07 PMOV33
5D41 PMOV34	5D4E PMOV35	5D86 PMOV40	5D83 PMOV45
5D9A PMOV46	5DA9 PMOV60	5DB4 PMOV61	5DB1 PMOV611
5E01 PMOV62	5E11 PMOV63	5E16 PMOV64	5E26 PMOV66
5E2E PMOV68	5E38 PMOV70	5E3D PMOV75	5E50 PMOV80

5D5F	PM0400	5D64	PM0405	5D6A	PM0410	5DD7	PM0625
5DED	PM0628	5CBA	PMOV10	5CDC	PMOV20	5CE4	PMOV25
5CE7	PMOV30	5CFC	PMOV31	5CF4	PMOV32	5D07	PMOV33
5D41	PMOV34	5D4E	PMOV35	5D86	PMOV40	5D83	PMOV45
5D9A	PMOV46	5DA9	PMOV60	5DB4	PMOV61	5DB1	PMOV611
5E01	PMOV62	5E11	PMOV63	5E16	PMOV64	5E26	PMOV66
5E2E	PMOV68	5E38	PMOV70	5E3D	PMOV75	5E50	PMOV80

SYMBOLS

D2:U.M65

5E5D	PMOV90	5C97	PMOVE	=D300	PORTA	6B69	PPH
6B5D	PPL	=D01B	PRIOR	=721D	PROGEND	4002	PSCOR0
4003	PSCOR1	4004	PSCOR2				
=00D4	R	=D20A	RANDOM	5E5E	RDSTAT1	5E8C	RDSTAT2
62EA	REDRAW	6C93	REFL10	6C9B	REFL20	6CA6	REFL30
6CA8	REFL40	6CAA	REFL50	6C91	REFLECT	6CC3	RFLPAT
4005	ROUND	428C	ROWBUF	7056	RSTD08	7140	RSTD10
7153	RSTD30	715E	RSTD31	7163	RSTD32	7165	RSTD35
7174	RSTD40	7178	RSTD44	7186	RSTD48	718A	RSTD50
712B	RSTDLI	=0012	RTCLOK				
4016	SAVBUF	696E	SCR233	6967	SCR242	696B	SCR244
400B	SCRCTR	5ADB	SCRFLG	4006	SCRFLG2	6935	SCR008
6946	SCR022	6957	SCR023	6958	SCR024	6980	SCR025
6983	SCR027	698E	SCR030	=00CA	SCROLG	400D	SCROLF
6926	SCROLL	400C	SCRSPD	=0231	SDLSTH	=0230	SDLSTL
=022F	SDMCTL	6870	SFX	6C12	SHAP05	6C16	SHAP10
6C2C	SHAP30	6C4A	SHAP40	6C05	SHAPEX	6C55	SHIF10
6C57	SHIF15	6C59	SHIF20	6C82	SHIF30	6C51	SHIFT
5EE1	SHOC03	5EEC	SHOC05	5F0B	SHOC07	5F14	SHOC08
5F21	SHOC10	5F3F	SHOC20	5F42	SHOC30	5EBA	SHOCAR
=D008	SIZEP0	=D009	SIZEP1	=D00A	SIZEP2	=D00B	SIZEP3
=D20F	SKCTL	=7E70	SMEM	4007	SNDCTL	6319	SPDCTR
6310	SREED	=0232	SSKCTL	67D2	SSTR1	67AE	SSTR2
67F6	SSTR3	6828	SSTR32	685A	SSTR4	6726	SSTR5
6762	SSTR5A	6788	SSTR5B	66CC	SSTR6	4C00	START
6F03	SWEEPR	6F03	SWEEPRM				
5800	TCHARS	51E2	TCHERA	646D	TERA10	644B	TERASE
5415	TFDLI	4001	TICS	5054	TIMER	4000	TIMRON
6505	TIN550	64FB	TIN551	6500	TIN552	64F6	TIN553
64AA	TINI10	64B4	TINI15	64B9	TINI20	64BB	TINI30
64C4	TINI40	64E4	TINI50	64F0	TINI52	650A	TINI54
6512	TINI56	6529	TINI60	653E	TINI62	6544	TINI64
654F	TINI66	6558	TINI70	656A	TINI74	656F	TINI80
65AA	TINI90	64A5	TINIT	65CC	TMAXSP	65D2	TMINSP
65D8	TPNDX	65AE	TPROB	5C16	TRA100	5AE6	TRAC02
5B01	TRAC05	5B09	TRAC06	5B13	TRAC10	5B38	TRAC18
5B3C	TRAC20	5B4D	TRAC30	5B5B	TRAC40	5B84	TRAC48
5B64	TRAC50	5B94	TRAC56	5B9F	TRAC60	5BA7	TRAC64
5BA9	TRAC65	5BA9	TRAC70	5BE0	TRAC80	5BED	TRAC90
5AE2	TRACK	=D010	TRIG0	664C	TRKDIE	6DC3	TRUCK
6E63	TRUCKM	4008	TRUCKX	4009	TRUCKY	62E0	TYPE
6B58	TYPREF						
=0001	UL	69B8	UPSC10	69C2	UPSC15	69D0	UPSC18
698F	UPSCR1	698F	UPSCRN	=0002	UR		
4D92	VBIN05	4E37	VBIN20	4E94	VBIN32	4E96	VBIN40
4E63	VBIN50	4E62	VBIN55	4D90	VBINT	=D40B	VCOUNT
68E5	VOICE1	=68E7	VOICE2	=68E9	VOICE3	=68EB	VOICE4
68ED	VPTR1	=68EF	VPTR2	=68F1	VPTR3	=68F3	VPTR4
=D405	VSCROL	68F5	VTIMR1	=0224	VVBLKD	=0222	VVBLKI
665C	VWDIE	663C	VWJUMP				
5F6B	WHEE05	5F73	WHEE10	5FAE	WHEE20	5F4B	WHEELS
5FAF	WHLPOS	5AE1	WHLSEQ	=D40A	WSYNC		
=000F	XCRSH	=E462	XITVBV	62F5	XPOS	62E9	XPRIME
=000E	YCRSH	62FE	YPOS	=000B	YTRKPT		